MAP working paper

Tanvi Jindal, Yili Luo

1   Introduction

Automobiles are arguably the most popular and important mode of transportation, with over 270 million vehicles registered in the United States alone. Data from National Highway Traffic Safety Administration (NHTSA), the Centers for Disease Control and Prevention (CDC), and the Federal Motor Carrier Safety Administration (FMCSA) show that car crashes are the leading cause of death for young people between ages 15 and 29, and more than 30,000 Americans die each year in vehicle accidents (**?**). To better understand vehicular collisions, scholars have developed numerous methods for measuring collision risk. Among the most popular are time-based methods, which are designed to capture how near or far a particular vehicle is from collision at any instant. In this manuscript we focus on time-based measures, beggining with a review of previous work, then proposing a new metric for general two-dimensional scenarios, and concluding with an application of our metric to data from a driving simulator experiment assessing vehicle-to-vehicle alert characteristics.

One of the most widely used methods of assessing collision risk is Time-To-Collision (TTC), which measures the "time required for two vehicles to collide if they continue at their present speeds and on the same path" (**?**). Since its inception, many extensions and refinements to TTC have been proposed. Brown (2005) adjusted the measurement in one-dimensional scenarios using the velocity at collision to resolve the limitation of TTC being zero at collision regardless of severity. **?** developed algorithms for calculating TTC in two-dimensional freeway settings, using different geometric shapes to represent the driver's vehicle and a buffer area around the vehicle to match the driver's perception on collisions. **?** studied right turning conflicts at intersections by grouping drivers by behavior and calculating TTC separately for each group. Nonetheless, these two-dimensional applications of TTC are limited to specific assumptions and scenarios, as a major limitation of TTC is that its values goes to infinity for non-collision trajectories. Circumventing this limitation without imposing restrictive scenario-specific assumptions is difficult, and to our knowledge there has not been a TTC-like collision measure that can be applied uniformaly to two-dimensional settings.

An alternative to TTC, developed by **?**, is Time of Closest Approach (TCA) measurement. Instead of relying upon intersecting vehicle trajectories, TCA is based upon the distance between vehicles as a function of time, with TCA representing the time at which the minimum distance occurs. By using the distance between the vehicles, TCA avoids the propensity of TTC to yield infiinite values when vehicle trajectories are disjoint. However, TCA cannot be used to uniformly compare across different scenarios as the implications of "minimum distance" are situation-dependent. Inspired by Brown's adjusted TTC, we decide to modify the calculation of TCA to produce a generalizable, time-based method of quantifying collisions in two-dimensions. Our approach estimates vehicles trajectories and calculates the degree by which the trajectories would need to be shifted in order for the vehicles to collide, with smaller shifts indicating the original trajectory was closer to collision than larger shifts. This provides a robust measure that can be generally applied to two-dimensional scenarios to assess safety or collision severity.

To illustrate the usefulness of our metric, we apply it data from a driving simulator experiment collected by the National Advanced Driving Simulator (NADS), using simulator inputs to quantify collision proximity at a two-dimensional intersection crossing where drivers must react to avoid colliding with a vehicle running a red light. In our application, we explore the effectiveness of using our measure to predict collision prior to impact, as well as potential advantages in statistical power that result from using our metric to retrospectively analyze experimental data.

2 Methodology

In this section we develop a metric for quantifying the proximity of two vehicles to collision. We begin with TCA as expressed by **?**, and add complexity to better mirror real world scenarios. Then, we adjust the calculation of TCA by introducing vehicle-specific latent variables, which are used to measure proximity to collision. Our development first considers vehicles as points, and later extends the method to consider vehicles as rectangles.

2.1 Point collisions

We first start by treating vehicles P and Q as point objects, introducing our notation using the diagram below:
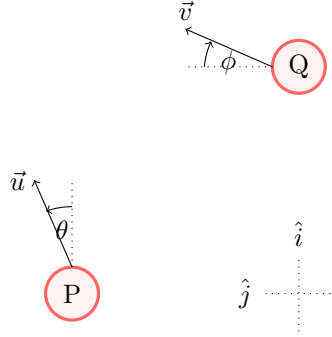
Here, $\hat{i}$ and $\hat{j}$ are axes of the vector plane, and $\vec{u}$ and $\vec{v}$ are the velocities of P and Q respectively. θ is the angle of rotation of $\vec{u}$ at point P from $\hat{i}$, and ϕ is the angle of rotation of $\vec{v}$ at point Q from $\hat{j}$, as illustrated. Notice, $\vec{u} = u_0 \cos(\theta)\hat{i} + u_0 \sin(\theta)\hat{j}$, and $\vec{v} = -v_0 \sin(\phi)\hat{i} + v_0 \cos(\phi)\hat{j}$.

For the purpose of this paper, we define anti-clockwise rotations to be positive and clockwise rotations to be negative. Therefore, with respect to illustration above, the angle θ is positive, and the angle ϕ is negative.

To calculate TCA for points P and Q, we express the displacement from P to Q as a function of time. The value of time which minimizes this function is the TCA of P and Q. Recognize that if P and Q collide with each other, their displacement at TCA will be 0. We express P and Q at time t, using the following notation:

$$P(t) = \begin{pmatrix} f_{\hat{i}}(t) \\ f_{\hat{j}}(t) \end{pmatrix}$$

where $f_{\hat{i}}(t) = \int_0^t u_x(t) \cdot dt$ and $f_{\hat{j}}(t) = \int_0^t u_y(t) \cdot dt$ where $u_x(t)$ and $u_y(t)$ are the $\hat{i}$ and $\hat{j}$ components of point P's velocity, and

$$Q(t) = \begin{pmatrix} g_x(t) \\ g_y(t) \end{pmatrix}$$

where $g_x(t) = \int_0^t v_x(t) \cdot dt$ and $g_y(t) = \int_0^t v_y(t) \cdot dt$ where $v_x(t)$ and $v_y(t)$ are the $\hat{i}$ and $\hat{j}$ components of point Q's velocity.

Thus the distance between P and Q at time t can be expressed as $\vec{d}(t) = P(t) - Q(t)$, which can be used to define the time to closest approach, $TCA = \text{argmin}_t\{\vec{d}(t)\}$.

2.1.1 Simple model

The simple model, as directly adopted from **?**, assumes the positions of P and Q at time t can be characterized by their initial positions P_0 and Q_0, and the constant velocity vectors $\vec{u}$ and $\vec{v}$:

$$P(t) = P_0 + t\vec{u} \tag{1}$$

$$Q(t) = Q_0 + t\vec{v} \tag{2}$$

The distance, $\vec{d}(t)$, is then given by:

$$\begin{aligned}
\vec{d}(t) &= P(t) - Q(t) \\
&= P_0 - Q_0 + (\vec{u} - \vec{v})t \\
&= s + (\vec{u} - \vec{v})t
\end{aligned} \tag{3}$$

where $s = P_0 - Q_0$.

TCA occurs when the magnitude of $\vec{d}(t)$ is minimum, requiring $d(t) = \sqrt{\vec{d}(t) \cdot \vec{d}(t)}$ to be minimized with respect to time.

$$\frac{d}{dt}d(t) = \frac{(\vec{u} - \vec{v}) \cdot (\vec{u} - \vec{v})t + 2s(\vec{u} - \vec{v})}{\sqrt{(\vec{u} - \vec{v}) \cdot (\vec{u} - \vec{v})t^2 + 2s(\vec{u} - \vec{v})t + s^2}}$$

Equating the derivative with provides a closed-form expression for the time to closest approach.

$$TCA = \frac{-s_0}{|\vec{u} - \vec{v}|^2} \tag{4}$$

2.1.2 Acceleration and Turning

The simple model enables a straightfoward calculation of TCA requiring relatively few inputs. However, this calculation is unable to reflect the important driver behaviors of braking, accelerating, and steering.

To modify the calculation to allow for variable velocity, we define the acceleration of P as $\vec{a_P} = a_{P0}\cos(\theta)\hat{i} + a_{P0}\sin(\theta)\hat{j}$, and define the acceleration of Q similarly. Incorperating acceleration into component of velocity yields:

$$\vec{u}(t) = (u_0 + a_{P0}t)\cos(\theta)i + (u_0 + a_{P0}t)\sin(\theta)\hat{j}$$
$$\vec{v}(t) = -(v_0 + a_{Q0}t)\sin(\phi)\hat{i} + (v_0 + a_{Q0}t)\cos(\phi)\hat{j}$$

To further modify the calculation to allow for turning we use the steering wheel angle to determine change in vehicle heading using the steering ratio. Because turn of the wheel is the angle by which the vehicle turns at every instant, it can be viewed as the derivative of a vehicle's angle of motion. Let turn of wheel of point P be W_P, then:

$$W_P = \frac{d\theta}{dt} \tag{5}$$

We assume W_P is constant, so upon integrating both sides of equation (5) with respect to t, we get the angle of motion as a function of time as follows:

$$\theta(t) = W_P \cdot t + \theta_0$$

where θ_0 is the angle at which P was turned at time $t = 0$.
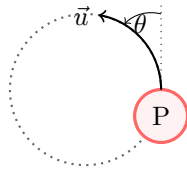
As the illustration indicates, given a fixed steering wheel angle, point P travels with velocity $\vec{u}$ in a circular motion. We point out that in most applications a complete revolution will not occur, and only the motion in the first quarter of the circle is relevent to the calculation.

2.1.3   Standardized Degree of Collision Avoidance (SDCA)

In the event that two vehicles do not collide, TCA indicates the time at which the distance between vehicles will be minimized based upon the set of driver inputs. However, it does not directly quantify the proximity of the two vehicles to a crash. Here, we define a *reaction* as a fixed set of inputs. We use this term because it might of particular interest to calculate TCA using vehicle trajectories upon having just realized the threat of collision. The basis of our metric, Standardized Degree of Collision Avoidance (SDCA), is to measure how delayed a driver's reaction would need to be in order for the scenario to result in a collision.

To approach this task, we begin by augmenting the expressions of $P(t)$ and $Q(t)$ by introducing two latent variables Z_P, which describes the reaction delay for P, and Z_Q, which describes the reaction delay for Q. This results in the adjusted positions functions $\tilde{P}(t)$ and $\tilde{P}(t)$:

$$\tilde{P}(t) = \left(\begin{array}{c} f_{\hat{i}}(t) + u_0 \cdot \cos(\theta_0) \cdot Z_P \\ f_{\hat{j}}(t) + u_0 \cdot \sin(\theta_0) \cdot Z_P \end{array} \right) \tag{6}$$

$$\tilde{P}(t) = \left(\begin{array}{c} g_x(t) - v_0 \cdot \sin(\phi_0) \cdot Z_Q \\ g_y(t) + v_0 \cdot \cos(\phi_0) \cdot Z_Q \end{array} \right) \tag{7}$$

Equating, $\tilde{P}(t) = \tilde{Q}(t)$ for all $t \in \mathbb{R}$, and solving for Z_P and Z_Q yields the combinations of reaction delays by P and Q that would lead to a collision. Thus, larger values of Z_P and Z_Q indicate a greater degree by which a collision was avoided. We next seek to summarize the combinations of Z_P and Z_Q to provide a single overall measure of collision avoidance.

Setting $\tilde{P}(t) = \tilde{Q}(t)$, implies that Z_P and Z_Q can be expressed functions of time such that each value of t corresponds to a pair $(Z_P(t), Z_Q(t))$. The pairs $(Z_P(t), Z_Q(t))$ can be viewed as a curve indexed by time, summarizing this curve amounts to summarizing all combinations of reaction delays that would have resulted in a collision. To summarize the curve, we calculate the expected distance, which we define as SDCA, from $(0,0)$ to the pairs $(Z_P(t), Z_Q(t))$.

$$\text{SDCA} = \frac{\int_{t_A}^{t_B} \sqrt{(Z_P(t) - 0)^2 + (Z_Q(t) - 0)^2} \cdot \sqrt{Z_P'(t)^2 + Z_Q'(t)^2} dt}{\int_{t_A}^{t_B} \sqrt{Z_P'(t)^2 + Z_Q'(t)^2} dt} \tag{8}$$

When performing this calculation, we restrict the curve to only include combinations occuring within the time indices defined by $t_A = \min(|Z_P|)$ and $t_B = \min(|Z_Q|)$. In effect, this restriction excludes (Z_P, Z_Q) combinations corresponding to large values of t where our motion projections are unlikely to still resemble the actual trajectories of each vehicle.

2.2   Object Collision

We move on to think of the vehicles as rectangles instead. Further, to calculate the metric, we start with treating the center of the rectangles as points P and Q described above. We get the illustration:
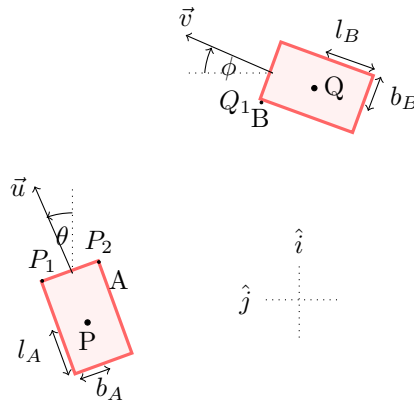
In order to determine the time to closest approach (TCA) of vehicles A and B, we undertake the following steps:

1) Determine the corner points of the vehicle.
Point P is determined as highlighted in section 2.1. To determine all the corner points of A, we add or subtract the $\vec{i}$ and $\vec{j}$ components of l_A and b_A of each. This gives the equation:
$$P_k(t) = \begin{pmatrix} f_{\hat{i}}(t) + (-1)^{k-k \bmod 3} l_A \cos\theta + (-1)^{\frac{k-k \bmod 2}{2}+1} b_A \sin\theta \\ f_{\hat{j}}(t) + (-1)^{k-k \bmod 3} l_A \sin\theta + (-1)^{\frac{k-k \bmod 2}{2}} b_A \cos\theta \end{pmatrix},$$
where k $\in \{1, 2, 3, 4\}$.
Note, θ lies in the direction of $\hat{i}$. The formula for Q_k is similar, but needs to be adjusted because ϕ lies in the direction of $\hat{j}$.

2) Find equations for a ll the edges of the vehicle.
Given the two corner points on an edge, we determine the vector equation of the line on which the edge lies. For example, the line joining P_1 and P_2 can be written as:
$A_k(t) = P_k(t) + (P_{k+1}(t) - P_k(t)) \cdot \lambda_k$
where $\lambda_k \in \mathbb{R}$ indicates position on A_k relative to P_k and $k + 1 = 1$ when $k = 4$.

3) Find the closest distance from each corner point of one vehicle to each edge of the other vehicle.
To find the minimum distance from a point Q_k to a line A_k, we first find the line passing through Q_k and perpendicular to A_k, and call it N_k. Then, we find the point of intersection m_k of A_k and N_k by equating A_k and N_k and solve for λ_k. Then, we plug the value for λ_k into A_k to get m_k.
The minimum distance between A_k and Q_k can be written as:

$$d_k(t) = \begin{cases} d(Q_k, m_k) & P_k \leq m_k \leq P_{k+1} \\ \min d(Q_k, P_k), d(Q_k, P_{k+1}) & otherwise \end{cases}$$

where d(a, b) indicates distance between a and b.

4) Minimize the closest distance for all points of both vehicles to get TCA
Each of the 32 point-line combinations has its own distance as a function of time. TCA is then found by finding the time at which minimum distance, across all combos, occurs.
In this example, we take $\min\{d_1, ..., d_{32}\}$ and differentiate the function with respect to time. Then, we equate the derivative with 0 to get the time of minimum distance, or TCA.

Like we did for point collisions, our goal is to adjust the metric to find the proximity to collision of vehicles A and B. We do this through the following steps:
1) Modify calculations to include delay from reaction time that would have led to a crash.
We modify our calculation of P(t) (equation 1) and Q(t) (equation 2) to create $\dot{P}(t)$ (equation 6) and $\dot{Q}(t)$ (equation 7), as defined before. We use these modified definitions to determine the corner points of A and B. Then, we undertake steps 1 to 3 illustrated above to get closest distance functions between every corner point and edge of A and B.

2) Find the minimum delay in reaction time for A and B for every time t.
For every time t, find the (Zp, Zq) combination closest to (0, 0) across all distances $\{d_1, ..., d_{32}\}$ using the function outlined in section 2.1.4. Thus, we select the minimum combined delay in reaction time of A and B at every point of time t.

3) Solve for SDCA.
As outlined in section 2.1.4, to get SDCA, we use the subset of (Z_P, Z_Q) such that it contains all combinations lying between the points where Z_P and Z_Q come closest to 0 respectively. Then, we calculate D (equation 8). Note, if $(Z_P, Z_Q) = (0, 0)$ for any value of t, SDCA is 0.

3 Application

We use simulator data collected by NADS to determine the validity of our metric as well as strengthen the argument made by NADS upon analyzing the data. Our SDCA metric serves two purposes:
1) to forecast occurrence of collision, and in case of non-collision, the extent of its avoidance based on reaction profile, and
2) to retrospectively analyze crash data. By testing the classification properties of the metric, we affirm the first purpose. By comparing the analysis of haptic alerts using SDCA with the results of the haptic study, we testify the second purpose.

3.1 Data Description

The data received from NADS consists of 180 participants, of which 156 participants had simulated time series data that was suitable for analysis. Each participant experienced a simulated crash scenario where the incursion vehicle was programmed to collide head-on with the participant's vehicle. Participants were given a haptic alert that began roughly 4 seconds before collision and the incursion vehicle became visible roughly 2.9 seconds before collision. A baseline was also established by giving 30 participants no haptic alert. The data was collected as a part of study on haptic warning characteristics and vehicle-to-vehicle (V2V) safety applications. The data is recorded as a time series, with increments of 0.0167 seconds. It was analyzed treating alert characteristics, consisting of inter-pulse interval (IPI), intensity (INT) and directional pattern (DIP), and alert levels as independent variables. The study applies a randomized block design to its experiment, treating INT= 50Hz, IPI= 0 and DIP= left only as default alert characteristic values. To determine the effect of any alert characteristic, its value is varied while the other two characteristics are set at their default values. This design creates 5 groups of 10 participants using randomized block assignment for the 3 characteristic in the haptic alert experiment.

3.2 Data Modification

The variables that we use in our analysis are vehicle speed, heading angle and a dummy variable indicating collision as a time series as well as alert characteristics and alert level by participant ID. We create scalar acceleration and turn of wheel by calculating rate of change of speed and heading angle at every instance of time. We take the reaction starting point to be the time after the start of the alert at which scalar acceleration or turn of wheel changes significantly. The threshold for a significant change is classified as a rate of change of 0.3^c for turn of wheel and 3 m/s^2 for scalar acceleration. Based on Hyden's proposal of driving behaviors performed to avoid collisions (1987), we split participants' reactions into five categories:
1. Scalar acceleration,
2. Scalar deceleration,
3. Turning,
4. Scalar deceleration and turning, and
5. No reaction.

The requirement for the reaction to fall into category 4 is for acceleration and turn of wheel to significantly change within 0.167 seconds. Given the category the initial reaction falls under, we determine the scalar acceleration and turn of wheel by taking its average from the start of the reaction to its maximum or minimum value.

Figure 1 is participant 65's scalar acceleration and turn of wheel in red and blue respectively for every time frame. The start of reaction is taken as the time at which turn of wheel first changes significantly, marked as First Reaction of Wheel Angle on the graph. Since scalar acceleration changes significantly within 0.167 seconds, participant 65's reaction is classified as 4. Then, we calculate the acceleration and turn of wheel by taking their average values from the start of reaction to their minimum and maximum values respectively. Doing so adds a lag in our SDCA calculation, but is necessary in estimating the participant's reaction profile more closely to their actual behavior.
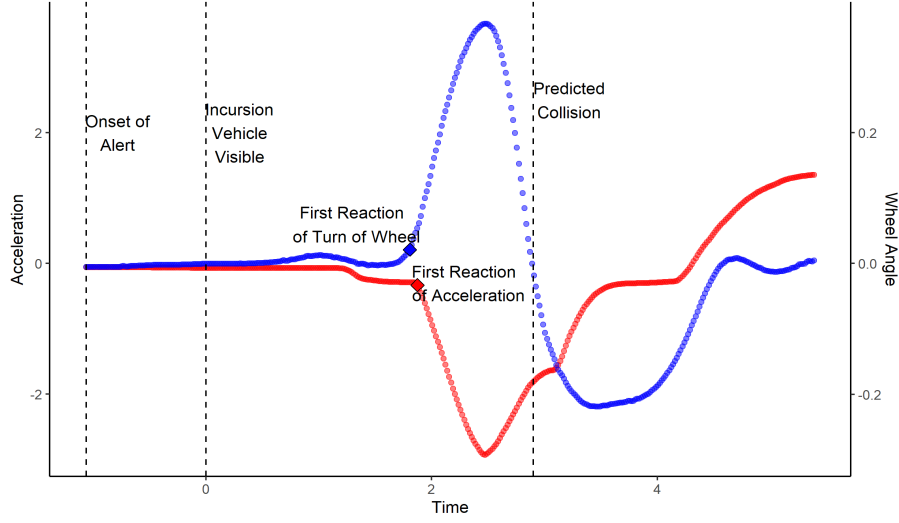
Figure 1: Time Series Data for Participant 65

3.3 Metric as a classifier

Using the time series data and our calculations of acceleration and turn of wheel, we solve for the proximity to collision as outlined in section 2.1.4. Upon iterating it over 156 complete cases, we get:
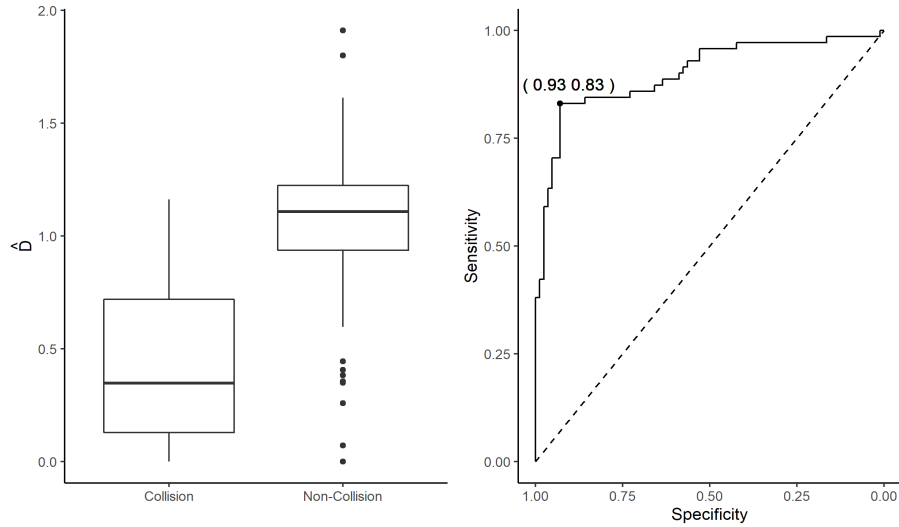

Figure 2: Boxplot and ROC Curve of $\hat{D}$ vs Collision

As demonstrated in figure 2, the boxplots are very separated for collision vs non-collision. This indicates that the SDCA values for the two groups are very different from each other. At a decision threshold of 0.90, we get 93% specificity and 83% sensitivity, which implies that, at D of 0.9, we can accurately predict 93% of collisions and only misclassify 17% of non-collisions as collisions. Additionally, the ROC curve indicates great accuracy of the metric. The AUC is 0.8911, which affirms that it is a good classifier for a range of decision thresholds.

3.4 Application on Haptic study

The purpose of the haptic study was to determine whether changing INT, IPI and DIP had a significant affect on a driver's reaction to collision. In the study, the reaction was categorized as collision, a dummy variable. We use distance as a measure for reaction and compare results for both.
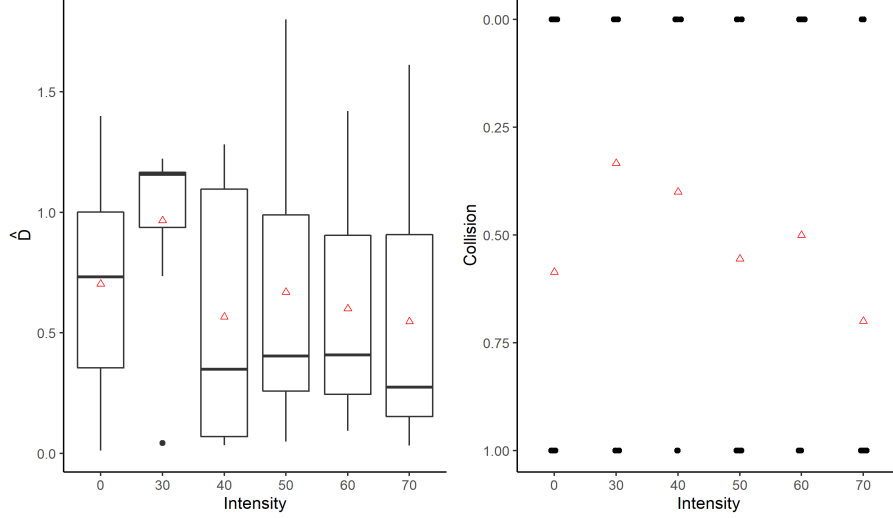

Figure 3: Trends in Alert Intensity using $\hat{D}$ and Collision

Through the two graphs in figure 3, we compare differences in response for different levels of intensity, with IPI and DIP set at their default values. The red triangles in both graphs in figure 3 indicate the mean value of the response variable for each level of intensity. Both graphs indicate that, compared to no alert, alerts at INT of 30Hz and 40Hz improve participant reactions to an imminent collision while alerts of higher INT levels minimally impact or worsen participant reactions.

Further, in order to measure the effects of different alert levels for each alert characteristic, we run a glm model on the data with non-null haptic alerts, using reaction as the response variable and INT, IPI and DIP as predictors.

Outcome	Family	Link	χ^2	df	p-value
$\hat{D}$	Gaussian	Identity	3.29	1	0.07
Collision	Binomial	Logit	2.42	1	0.12

Table 1: Likelihood ratio test results for INT

Depicted in table 1 are comparisons of likelihood ratio tests for INT applied to linear and logistic regression models with INT, IPI and DIP as predictors and D and collision as response variables respectively. Both models pick up the effect of INT in the same direction, but the statistical confidence in their claims differ. While INT has a p-value of 0.12 with collision as the response, INT has a p-value of 0.07 with expected distance as the response. The p-value reduces by a factor of 2, making INT statistically significant at a 10% threshold. While not significant, similar patterns are observed with IPI and DIP. These improvements indicate that using expected distance captures trends observed in the data for all alert characteristics better than collision. We believe that the added information of expected distance helps detect the effects observed visually. Hence, our metric improves retrospective analysis of crash data, fulfilling its second purpose.

4   Discussion

TCA is the time to closest approach of two vehicles with set reaction profiles and a fixed reference point. In the event of a collision, TCA is an effective method of determining the time at which a collision occurs relative to the start of the reaction. Therefore, in case of collisions, smaller values of TCA indicate greater imminence of the collision. However, we have three salient criticisms of the metric:

1. Comparing across different crash data poses challenges as the reference point across all data needs to be fixed in order to get a meaningful interpretation of TCA for the data. This is a substantial limitation, as reference points are not generally consistent across real-world crash situations.

2. TCA has limited applicability in the absence of a collision across multiple scenarios, as it is not reflective of imminence of danger. For example, a TCA of 4s does not distinguish between two sets of vehicles that are 5m and 15m apart from each other at TCA. Therefore, TCA does not uniformly reflect proximity to collision.

3. In case of a collision, TCA does not include information about severity of a collision and hence, does not distinguish between major and minor collisions.

SDCA addresses these limitations. SDCA, or standardized degree of collision avoidance can be understood as the proximity to collision of two vehicles. SDCA turns the binary variable collision into a spectrum of positive values. By design, values farther away from 0 indicate greater levels of safety from collision.
SDCA is capable of quantitatively measuring the degree of collision (or non-collision) between two vehicles with set reaction profiles. This solves for our second and third criticisms of TCA. SDCA naturally equalizes different reference points because it uses delay in reaction times, rather than the time from the reference point itself. For example, if two vehicles are on track to collide and are currently two seconds away from their collision point their TCA is 2s while SDCA is 0. If we change the reference point to when the two vehicles are only 1s away from each other, their TCA becomes 1 while the SDCA remains 0. This simple example illustrates how SDCA is not reference-point dependent, which allows different scenario to have their own reference points while still providing for meaningful comparisons across scenarios.
In Sec (ref-methods) we present two separate SDCA calculations, one treating vehicles as points and another treating them as rectangles. We argue that both have utility.
The first method is point based calculation of SDCA. We conduct our analysis using this method, Each point used is located in the center of one of the two vehicles. $\hat{D} = 0$ in this interpretation would indicate a collision of the centers of the vehicles, the most severe collision scenario theoretically possible. Since SDCA comprises of a spectrum of positive values, increasing values indicate decreasing severity of collision, and after a point, decreasing proximity to collision. In applying the metric to data, we set data-dependent thresholds to interpret the extent of collisions and non-collisions for different participants. As illustrated in figure 4, the values in SDCA can be interpreted according to their position in the spectrum.

As figure 1 indicates, the boxplots of $\hat{D}$ for collision and non-collision differ greatly. We use their mean values as thresholds to interpret different SCDA values obtained. Object based calculation- All non-zero values of SCDA signify non-collision, with increasing values indicating less proximity to collision. If SCDA is 0, we can identify the point-edge pair that collides by observing the distance function that first returns 0. However, this information is not encoded in SCDA itself but in its calculation. Both versions of SCDA have their strengths. Point based calculation of SCDA gives a range of values that determine severity and proximity of collision, which makes it a superior method for data analysis. Conversely, object based calculation of SCDA has potential of revealing the position of crash. Also, since the object based calculation sets a clear

threshold of non-collisions (all non-zero values), it might do a better job at forecasting collisions than point based calculations of SCDA.

This metric is good at forecasting the extent of collision and non-collision that a set of vehicles encounter, as illustrated in section 3. There are multiple ways to use SDCA for forecasting.

One method follows the calculation illustrated in section 3.2. The method requires estimation of reaction profiles at a fixed start point for a set of vehicles and use of the estimated profile to calculate SDCA. In real world scenarios for forecasting collision from a fixed start point, there is a lag in our metric to get accurate estimates. This lag is still shorter than the length of reaction, and some forecasting power is sacrificed, but not lost. The advantage of using this method is that, as only a single SDCA value is obtained for a set of vehicles, the estimated SDCA is relatively simple to calculate and interpret for a scenario. Hence, it allows for comparisons across different V2V collision scenarios, as depicted in section 3.3. SDCA estimated through this method is also a good approximation of the true SDCA if the vehicles' reactions do not alter significantly.

Another method for calculation of SDCA is to treat every instance of time after the imminence of the threat of collision as a separate reference point and determine the vehicles' reaction profiles instantaneously. This makes SDCA a dynamic measure that forecasts the occurrence of collision based on a vehicle's immediate reaction profile. This method is optimal for real-world collision avoidance as it reflects the threat of collision posed to a vehicle at every instance. Hence, a driver can modify their reaction profile repeatedly to obtain larger SDCA values and avoid a collision. However, because this method gives a series of SDCA values for every time frame, its interpretability for a scenario is limited. This hinders comparisons across multiple V2V scenarios as well.

As mentioned above, using object based calculation of SDCA might be better at forecasting as it considers all points of collision between two vehicles. By having a clear threshold that separates collisions and non-collisions, it can more accurately help vehicles avoid collisions entirely. Conversely, if a collision is imminent, point based calculation of SDCA would be useful as it consists of a spectrum of values that separate severe collisions from mild collisions. In a real-world situation, therefore, both calculations of SDCA should be used in tandem.

As demonstrated in section 3.4, SDCA improves statistical analysis, as compared to a dummy variable for collision. By using a continuous variable, we are able to distinguish between collisions of different levels of severity and therefore get a stronger analysis. For example, a head-on collision and a mild scrape would both be classified as a collision, when their levels of severity differ greatly. This distinction is visible using SDCA, making it a more robust response variable. Figure 3 and table 1 highlight the superiority of SDCA to a binary collision variable, which depict similar trends for INT across both response variables, but produce more statistically significant results with SDCA as the response. Using the point wise calculation of SDCA is better for analysis because it incorporates that distinction for both, collisions and non-collisions.

5   Conclusion

We create time-based metric that measures proximity to collisions in real-world, we adapt time to closest approach (TCA) by Schwarz to create the standardized degree of collision avoidance (SDCA). SDCA is a measure of proximity of collision, and converts the binary variable collision into a range of values. We use point wise and object wise calculations to determine SDCA. Both methods have different applications. SDCA can be used to forecast the occurrence or non-occurrence of collision as well as replace the binary collision variable to strengthen analysis.