Estimation of Road Transverse Slope Using Crowd-Sourced Data from Smartphones

Abhishek Gupta[†], Abhinav Khare[†], Haiming Jin[§], Adel Sadek[†], Lu Su[†], Chunming Qiao[†]
[†]State University of New York at Buffalo, Buffalo, NY, USA
[§]Shanghai Jiao Tong University, Shanghai, China

Abstract

Integration of information on road transverse geometric features such as cross slope and superelevation in digital maps can widen the scope of its applications, which is primarily navigation, by enabling driving safety and efficiency applications such as Advanced Driver Assistance Systems (ADAS). The huge scale and dynamic nature of road networks make sensing such road geometric features a challenging task. Traditional methods oftentimes suffer from high cost, limited scalability and update frequency, as well as poor sensing accuracy. To overcome these problems, we propose a cost-effective and scalable road transverse slope estimation framework using sensor data from smartphones. Based on error characteristics of smartphone sensors, we intelligently combine data from accelerometer, gyroscope and GPS to estimate road transverse slope profile of a road segment. To improve accuracy and robustness of the system, the estimations of road transverse slope from multiple sources/vehicles are crowd-sourced to compensate for the effects of varying quality of sensor data from different sources. Extensive experimental evaluation on a test route of 9km demonstrates the superior performance of our proposed method, achieving 350% improvement on road transverse slope estimation accuracy over existing methods, with 90% of errors below 0.5°.

1 Introduction

The transverse geometric features of road surface, namely cross slope and superelevation, play a crucial role in ensuring road safety. Cross slope is defined as the transverse slope angle of the road with respect to the horizon. Cross slope provides a drainage gradient so that water can run off the surface to a drainage system such as a street gutter or ditch [25] (Fig. 1). In horizontal curves, the cross slope is gradually banked into superelevation angle to reduce steering effort and lateral force required to go around the curve [25] (Fig. 1). Rich information on transverse road geometry features such as cross slope and superelevation plays a crucial role in risk assessment and engineering design of road segments. For e.g. adequate superelevation is highly important to allow the vehicles to safely negotiate horizontal curves. If not properly designed, inadequate superelevation can potentially lead to serious roadway departure crashes. Similarly, improper cross slope can cause a vehicle to drift and skid laterally while braking [4]. Moreover, information on transverse road slope assist in tasks such as accident reconstruction and investigation [2]. Such information is crucial for transportation agencies to take adequate measures (for e.g. road surface treatment, road geometry improvement, etc.) to reduce the crash risk associated with a road segment [16].

As the automobile sector is gradually moving towards complete autonomy, there is an ever-increasing requirement for scalable and cost efficient solutions to collect rich and accurate data about road networks. Especially, gathering and estimating accurate road transverse geometric features can play a crucial role in enabling Advanced Driving Assistance Systems (ADAS) [27], which helps in improving driving safety and efficiency. For e.g., a **"Curve Speed Warning System"** [22] assesses threat levels for a driver approaching a curve too quickly using information on transverse road geometry features. Other applications of information on transverse road geometry features include terrain based localization [3], optimal control in autonomous vehicle [18], scalable creation of HD Maps [10], etc.

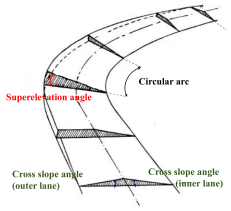

Figure 1: Cross slope and superelevation design.

The task of estimating road geometry features is challenging due to the sheer scale of the road networks, as well as their dynamic nature stemmed from the construction of new roads and maintenance of the existing ones. According to US Department of Transportation (USDOT), between 2000 and 2016, the U.S. built an average of 30,427 lane miles of roadway per year [23]. In addition to new roads, existing roads are maintained/modified for accident vulnerability prevention, traffic flow enhancement, etc. These maintenance tasks often result in change in road geometry features [16]. Typically, estimation of road transverse slope is done by extensive surveys conducted by instrumented vehicles [5, 17]. However, high deployment cost due to expensive equipment/sensors and dedicated labor, renders these approaches difficult to scale. An ideal sensing framework for the task of road geometry features estimation should be scalable, cost-efficient, and capable of providing frequent data updates. Also, the system should be accurate enough to enable the aforementioned applications such as ADAS and Autonomous driving.

Due to their ubiquity and rich array of onboard sensors (IMU, GPS and Magnetometer), smartphones provide a unique opportunity for developing a cost-efficient and scalable crowd sourcing solution for road transverse slope estimation. Assuming that a smartphone is stationary inside a moving vehicle, the 3D orientation estimates of pitch, yaw and roll of the smartphone can yield information about geometry of the road the vehicle is traveling on. However, use of smartphones for the task of road transverse slope estimation introduces the following challenges : **a) Noisy**

smartphone data: Commodity smartphones' MEMS sensors are prone to biases and drifts, making smartphone data noisy and error-prone. Furthermore, in a driving environment, the drifts can be dynamic in nature and thus change over time due to factors such as linear acceleration, road conditions, temperature, etc. [8, 30]. **b) Arbitrary smartphone placement and orientation:** Designing a practical and easily deploy-able crowd sourced system imposes the requirement of keeping user/participant involvement to the minimum. Therefore, we work with the assumption that the smartphone can be placed in any arbitrary orientation and position inside the vehicle. However, as discussed in the paper, imperfections in coordinate alignment between vehicle's and smartphone's frame of reference can cause biased estimation of transverse slope profile. **c) Varying QoI of data from different sources/vehicles:** Heterogeneity introduced due to factors such as varying quality of sensors in different smartphones, variable physical properties of vehicles (e.g. suspension properties), variable quality of phone-holders, etc. can result in varying QoI (Quality of Information) from different sources/vehicles, which might influence the estimations of transverse road slope. The above mentioned factors can cause significant errors in estimation w.r.t the application of road transverse slope estimation, where the margins of error are less (typically, values of cross slope/superelevation are $< 6^{\circ}$).

In this paper, we propose a novel and easily deployable solution to estimate transverse slope profile of a road segment using crowd-sourced data from smartphones. The proposed solution leverages acceleration and angular velocity data from accelerometer and gyroscope of the smartphone and vehicle speed data from GPS. To handle the challenge of noisy smartphone data, we propose a novel strategy to fuse estimations from gyroscope and accelerometer guided by our understanding of error characteristics of smartphone's sensors. In particular, the gyroscope is precise in capturing the shape of the road, but suffers from error accumulation/drift for estimation over long periods of time. Also, the drift is unpredictable in nature and can change over the course of a driving trip. On the other hand, accelerometer is not prone to drift, but is susceptible to large errors and biases which are typically correlated to the dynamics of the vehicle and errors in coordinate alignment process. Based on this observation, we propose to use gyroscope as the primary sensor for road transverse slope profile estimation. Accelerometer, on the other hand, is used to opportunistically provide chosen "anchor snapshots", which are used to correct drift of gyroscope. Furthermore, to handle the challenge of bias in estimations introduced due to inaccuracies in coordinate alignment, we design a novel strategy that fuses observations of superelevation from vehicle kinematics model and road design principles. Finally, we aggregate data from multiple sources/vehicles to improve accuracy and robustness of the system by handling varied quality of sensor data from different sources. Extensive evaluation of our proposed method on a $\approx$ 9km route, using natural driving traces from smartphones shows the possibility of achieving accuracy comparable to that of high-cost specialized instrumented vehicles. The system achieves 90% error less than 0.5° in estimating road transverse slope profile and outperforms existing approaches by a considerable margin.

2 Related Work

Traditionally, accurate road geometry features acquisition is typically done using survey vehicles instrumented with high-quality sensors such as lidars, laser scanners, cameras and high-end IMU's, etc. [5, 17, 21]. In some cases, remotely sensed data from satellite or aerial imagery (e.g. aircraft mounted with Lidar) is used to estimate road geometry features [19]. Remote sensing approaches are capable of providing large scale estimations but, suffer from low resolution and accuracy. Furthermore, due to high deployment and collection cost, all the above mentioned approaches are limited in their ability to scale and provide frequent data updates.

To alleviate high cost, some recent methods proposed to use smartphone as a sensing platform to estimate transverse road slope profiles [20, 28]. [28] uses vehicle kinematics model to compute point estimates of superelevation on horizontal curves. The above method however, is limited as it does not generate continuous transverse slope profile of a road segment, which is crucial to enable applications such as localization [3], autonomous driving [18], etc. Moreover as discussed and evaluated in the paper, superelevation estimation using vehicle kinematics model can be prone to errors due to factors such as noisy GPS data, driving maneuvers, etc. [20] uses a complimentary filter to add/fuse estimate from accelerometer and gyroscope to estimate tilt of vehicle w.r.t road surface, which is added to superelevation estimated using vehicle kinematics model to derive transverse slope profile of a road segment. However, it has been shown that directly adopting traditional sensor fusion techniques such as complimentary and kalman filter on smartphones results in poor performance [30]. As we explore in the evaluation section, [20] produces sub-optimal results predominately because of its inability to counter errors induced by unpredictable nature of gyroscope drift and the high accelerometer noise in a dynamic driving conditions.

In addition to challenges introduced due to noisy smartphone data, inaccuracies in coordinate alignment of IMU's can result in erroneous estimation of orientation [17]. As we discuss later in paper, biases in estimations due to inaccurate coordinate alignment between smartphone's and vehicle's frame of references can result in significant errors in estimation of road transverse slope profile. Previous methods solve this issue by manually placing the phone in a known orientation inside the vehicle [20, 28]. However, such manual involvement is not desirable for a crowd-sourced system, as it might discourage the participants from performing the actual task. Finally, the existing methods do not account for varying Quality of Information (QoI) from heterogeneous sources. Varying QoI might arise from factors such as varying quality of sensors on different smartphones, varying physical properties of vehicles (for e.g. suspension properties), varying quality of phone-holders, etc., which might influence the estimation of transverse slope profile of road.

3 Road Transverse Slope Profile Estimation Framework

The IMU sensors on a smartphone fixed inside a moving vehicle will be able to capture the dynamics of the vehicle, which is typically a result of the forces induced by driver control (acceleration/braking done to achieve desired speed and steering control

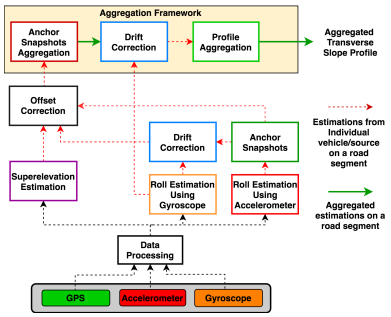

Figure 2: System Architecture.

for lane changes on a straight road), as well as the road's horizontal (steering control while negotiating a turn) and vertical (while going up-hill or down-hill a road/while travelling on a road with super-elevation) geometry. Therefore, if the contribution of driver control is removed from the smartphone signal, we will be left with a signal encapsulating road geometry information. The proposed approach for road transverse slope estimation from a smartphone is based on the above intuition.

Fig. 2 illustrates the architecture of our proposed system for road transverse slope profile estimation. The raw data from the smartphone is pre-processed in the "Data Processing" module, as described in Sec. 3.1. Initial estimates of transverse slope are computed using data from gyroscope and accelerometer in "Roll Estimation Using Gyroscope" and "Roll Estimation Using Accelerometer" modules, respectively, as described in Sec. 3.2. The "Anchor Snapshots" module opportunistically selects roll estimation during stable driving phases, to handle error prone estimations from accelerometer due to unstable vehicle dynamics, as described in Sec. 3.3.1. The anchor snapshots are then fused with the roll estimations from gyroscope in the "Drift Correction" module, to compensate drift associated with gyroscope estimations, as described in Sec. 3.3.2. "Superelevation Estimation" module computes superelevation on horizontal curves leveraging vehicle kinematics model. These estimates of superelevation are used to correct the bias/offset associated with anchor snapshots in the "Offset Correction" module. The "Aggregation Framework" handles varying QoI of data from different sources/vehicles. Specifically, the estimations of anchor snapshots (after bias/offset correction) from an individual vehicle/source are aggregated in "Anchor Snapshots Aggregation" module. The aggregated anchor snapshots are used to compensate the drift associated with roll estimations using gyroscope. These drift compensated cross slope profiles from an individual vehicle/source are then aggregated in the "Profile Aggregation" module to estimate a single road transverse slope profile of a road segment, as described in Sec. 3.5.2. Next, we will present the integrated components of our design.

3.1 Data Processing

3.1.1 Preprocessing Due to their relatively low quality, smartphone sensors tend to output data more prone to noise, which is further amplified by vibrations of the vehicle. Therefore, we smooth the signal from accelerometer and gyroscope by passing it through a second-order Butterworth low-pass filter. The accelerometer and gyroscope are sampled at 200Hz, whereas the velocity data from GPS arrives at a much lower rate of 1Hz. To perform trace alignment, we thus interpolate velocity data to get 200 samples/sec. We also employ trace synchronization [24] to prevent data from the IMU and GPS of the Smartphone to go out of sync. Finally, we segregate data for different road segments on the test route (Fig. 12). We divide the test route into road segments based on presence of an intersection or a stop sign.

3.1.2 Coordinate Alignment To sense meaningful dynamics of the vehicle using a smartphone, it is necessary to align the phone's coordinate system with the vehicle's. We will work with the coordinate system shown in Fig. 3 in this paper. We leverage an existing technique to perform the alignment [24], which results in a 3×3 rotation matrix R_{PC} for transforming the smartphone's data to the vehicle's frame of reference. The first, second and third columns of R_{PC} are unit vectors $\vec{x_u}$, $\vec{y_u}$ and $\vec{z_u}$ pointing in the direction $+X_C$, $+Y_C$ and $+Z_C$ respectively as also illustrated in Fig. 3. $\vec{z_u}$ is estimated when the vehicle is stationary and thus, the acceleration reported by the accelerometer constitutes gravity due to earth. We estimate $\vec{z_u}$ at the beginning of a trip, when the vehicle is on a level surface (e.g., in a parking lot). $\vec{y_u}$, which points in the moving direction of the vehicle, is estimated when the vehicle undergoes acceleration on a straight road segment (e.g., when the vehicle starts from zero velocity after stopping at an intersection). Finally, $\vec{x_u}$ is the cross product of $\vec{y_u}$ and $\vec{z_u}$. Coordinate alignment is done once, during a trip, as soon the system captures a valid acceleration profile for the estimation of $\vec{y_u}$. We also use *pitch*, *yaw*, and *roll*, to denote the vehicle's rotations about the lateral axis X_C, perpendicular axis Y_C, and longitudinal axis Z_C, respectively.

3.2 Vehicle Roll Estimation Using Gyroscope and Accelerometer

Varying 3D road geometry results in rotation of vehicle in the vertical Y_C-Z_C and X_C-Z_C planes (Fig. 3). The smartphone fixed in the vehicle also undergoes changes in orientation accordingly. Specifically, varying road transverse slope will result in rotation of the vehicle about the Y_C axis, thus contributing to the change in the vehicle's roll. We discuss our multimodal estimation methodologies as follows.

3.2.1 Roll Estimation Using Gyroscope A smartphone's gyroscope reports the real-time angular velocities $\omega_{X_p,t}$, $\omega_{Y_p,t}$, and $\omega_{Z_p,t}$ around the X_p, Y_p and Z_p axis of the smartphone, respectively. Assuming that gyroscope data has been aligned with vehicles coordinate frame, the roll at time instant t, ($\phi_{gyro,t}$) can be estimated by integrating the angular velocity of the vehicle about Y_C axis ($\omega_{Y_c,t}$) using Eq. 1.

$$\phi_{gyro,t} = \phi_{gyro,t-1} + \omega_{Y_c,t}.\Delta t \tag{1}$$

3.2.2 Roll Estimation Using Accelerometer The acceleration captured by a smartphone's accelerometer contains both the vehicle's longitudinal/lateral accelerations and the gravity. When a vehicle is stationary, the measured acceleration of the smartphone is only due to gravity. When the vehicle is moving, road transverse slope will result in rotation of vehicle around Y_c, contributing to roll of the vehicle. This will result in rotation of the gravity vector. Therefore,

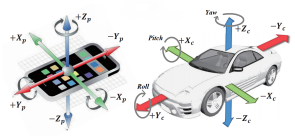

Figure 3: Smartphone and vehicle Coordinate System.

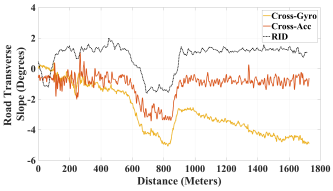

Figure 4: Vehicle roll estimation using Gyroscope and Accelerometer.

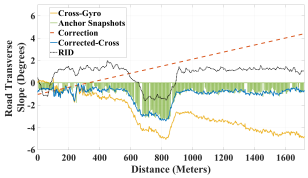

Figure 5: Opportunistic transverse slope profile estimation.

roll of the vehicle can be estimated by tracking gravity vector's rotation in the plane determined by $\vec{x_u}$ and $\vec{z_u}$, where $\vec{z_u}$ is estimated when the vehicle is stationary on a level surface during coordinate alignment process.

However, when the vehicle is moving, accelerometer captures longitudinal (braking and accelerating of the vehicle) and lateral acceleration (when the vehicle is negotiating a turn/lane change), along with gravity. Therefore, to estimate gravity, contribution of longitudinal and lateral acceleration has to be removed from accelerometer readings. We leverage speed information of a vehicle from the GPS device on the smartphone to estimate longitudinal and lateral acceleration of the vehicle. Longitudinal acceleration magnitude (A_t) of the vehicle is estimated by differentiating the speed and projecting it on Y_C using Eq. 2. Lateral acceleration of the vehicle is estimated using velocity V_t and angular velocity about Z_c ($\omega_{Z_c,t}$) using Eq. 3.

$$\vec{A_{Y_c,t}} = A_t.\vec{y_u} \tag{2}$$

$$\vec{A_{X_c,t}} = (\omega_{Z_c,t}.V_t).\vec{x_u} \tag{3}$$

When the vehicle is moving, the gravity vector $\vec{G_t}$ can be estimated using Eq. 4 where, $\vec{A_t}$ is the acceleration vector reported by the smartphone, $\vec{A_{Y_c,t}}$ and $\vec{A_{X_c,t}}$ are longitudinal and lateral acceleration of the vehicle.

$$\vec{G_t} = \vec{A_t} - \vec{A_{Y_c,t}} - \vec{A_{X_c,t}} \tag{4}$$

Finally, roll estimation using accelerometer ($\theta_{acc,t}$) is done by tracking rotation of gravity vector in the plane determined by $\vec{x_u}$ and $\vec{z_u}$ (Eq.5).

$$\phi_{acc,t} = \angle(\vec{G_t}, \vec{x_u}) \tag{5}$$

3.2.3 Error Characteristics of Gyroscope and Accelerometer Fig. 4 illustrates the estimated transverse slope profile of a road segment using gyroscope (labeled "Cross-Gyro") and accelerometer (labeled "Cross-Acc"). We draw the following observations on the error characteristics of estimations using gyroscope and accelerometer.

- **Gyroscope :** Estimations from gyroscope capture the shape of the road profile accurately. However, the estimations suffer from drift (Fig. 4). Moreover, as explored in [8], the drifts can be dynamic in nature and can vary both in magnitude and direction.
- **Accelerometer :** On the other hand, estimation using accelerometer is not prone to drift. However, the signal exhibit large errors with high variance. These errors are majorly attributed to pollution of the accelerometer signal by forces induced due to driver control events [8]. For e.g. lane changing events at ≈ 300 meters in Fig. 4 result in large disturbances in the estimation using accelerometer. Moreover, the estimation from accelerometer is characterized by a bias/offset which is primarily attributed to errors in the coordinate alignment (Sec. 3.1.2).

Deriving insights from the aforementioned observations on the error characteristics of gyroscope and accelerometer, we design a sensor fusion strategy that exploits the complimentary nature of the two sensors to estimate road transverse slope profile.

3.3 Opportunistic Transverse Slope Profile Estimation

As explored in the previous section, gyroscope and accelerometer have complimentary properties w.r.t the task of road transverse slope estimation. Traditional sensor fusion techniques such as kalman and complimentary filters can exploit sensor redundancy to improve accuracy. For optimal performance, these fusion techniques require estimates of sensor's noise/error characteristics, which are fed as parameters to the fusion framework. However, as mentioned in the previous section, the quality of smartphone sensor's response can be unpredictable in nature. Thus, in dynamic environments, it is difficult to optimize parameters for such fusion techniques to perform optimal error control. Therefore, directly adopting such sensor fusion approaches for the task of road transverse slope estimation produces sub-optimal results, as demonstrated in the Sec. 4.

Based on our understanding of error characteristics of accelerometer and gyroscope in different conditions, we propose an opportunistic framework for road transverse slope estimation. The underlying intuition is as follows :

- We treat gyroscope as the "primary" sensor. This is based on the observation that estimation using gyroscope is precise in capturing the 3D shape of the road segment accurately. Also, the estimation is not prone to errors induced by varying vehicle dynamics.
- To compensate for the drift-induced errors in estimation from gyroscope, we leverage estimation from accelerometer as an anchor for correcting the drift. Specifically, we opportunistically select "anchor snapshots" during "stable" driving phases to correct drift of the signal from gyroscope.
- To counter unpredictable response of smartphone's gyroscope, drift correction is done "on the go". Specifically, we perform drift correction in windows of given distances.

Next we present our proposed framework in more detail.

3.3.1 Capturing Anchor Snapshots As previously discussed, road transverse slope estimation using accelerometer is error prone due to forces induced by vehicle dynamics. Constant or frequent acceleration/braking and turning/lane changes are the cause of these forces. Even if we remove the contribution of vehicle dynamics,

as described in Sec. 3.2.2, the method is still not capable of fully compensating for the contribution of vehicle dynamics mainly because of the noise in GPS velocity data, thus resulting in inaccurate estimation of longitudinal and lateral acceleration of the vehicle. Because of the above, the error of transverse slope estimation using accelerometer is magnified when the dynamics of vehicle are not stable, especially during rapid lateral acceleration/deceleration phases.

Keeping the above in mind, we opportunistically filter out the transverse slope estimations during stable driving phases based on the following two metrics :

- **Longitudinal Acceleration**: We reject the estimations during "high" acceleration/deceleration phases. These phases are filtered out based on $longAcc_{thresh}$.
- **Longitudinal and Lateral Jerk** : We also reject the estimations when the longitudinal/lateral dynamics of the vehicle are unstable (e.g., rapid acceleration/deceleration phases, lane changing events, etc.). This is achieved by rejecting estimations based on $longJerk_{thresh}$ and $latJerk_{thresh}$, where jerk is the rate of change of acceleration.

The transverse slope estimations during stable driving phases are segregated into bins of length 2m. Finally, observations in each bin are averaged to estimate the anchor snapshots. Fig. 5 illustrates the estimated anchor snapshots using the approach described above. The optimal value of $longAcc_{thresh}$, $longJerk_{thresh}$ and $latJerk_{thresh}$ are empirically driven and set to $1.5m/sec^2$, $0.3m/sec^3$ and $0.2m/sec^3$, respectively.

3.3.2 On the Go Drift Correction The anchor snapshots estimated in the previous section are used to estimate the gyroscope drift (Fig. 2, middle). We use the least-squares [29] method to fit a line to the difference between the transverse slope estimation from gyroscope and the anchor snapshots. To account for the dynamic nature of gyroscope drift, the correction is done in windows of distances of size $wind_{drift}$, as shown in Eq. 6, where R_L is the length of the profiled road segment and D_{anch} is the list of distances between consecutive anchor points on the road segment.

$$wind_{drift} = \max((R_L/3), \max(D_{anch})) \tag{6}$$

Fig. 5 illustrates the result of on the go drift correction mechanism. It can be seen from the figure that the proposed mechanism takes care of gyroscope drift. However, due to the bias/offset present in estimation from accelerometer, the corrected road transverse slope profile (labeled "Corrected-Cross") also exhibits an offset. In the next section, we describe our methodology to mitigate the problem of the offsets.

3.4 Offset Correction

The coordinate alignment methodology described in Sec. 3.1.2 computes transformation matrix between smartphone's and vehicle's frame of references. The method assumes a level surface to estimate the vertical ($\vec{z_u}$) and lateral ($\vec{x_u}$) axis of the vehicle. Due to the alignment of vertical axis (Z_c) of vehicle with the direction of gravity (vertical axis in earth's frame of reference) on a level surface, vehicle roll estimated using Eq. 5 would estimate the transverse slope of the road surface correctly. However, estimation of $\vec{z_u}$ on a level surface might not always be possible. While calculating $\vec{z_u}$ when the vehicle is stationary, the road surface more often than not will have some inclination due to grade or cross slope or both. Hence, estimation of $\vec{z_u}$ on a non-level surface will incorporate tilt of the surface of the road w.r.t earth.

A solution to mitigate the offset problem can be to perform manual coordinate alignment of the smartphone i.e., fixing the smartphone in a known orientation w.r.t the vehicle. For e.g. as done in [20], the smartphone is fixed such that its vertical plane is aligned with the vehicle's vertical plane. However, such manual involvement of participants is not desirable in a crowd-sourced system.

With the aim of designing a crowd-sourced system transparent to participants, we leverage superelevation observations on horizontal curves made using vehicle kinematics model as "fixes" for correcting the bias/offset associated with the estimated transverse slope profile. The key insight is that the estimations made using vehicle kinematics model can provide a reasonable observation on the"true" superelevation of a horizontal curve. However, superelevation estimations from vehicle kinematics model can be error prone due to factors such as noisy GPS data, driving maneuvers, etc. as discussed in more detail in the next section. To handle these noisy superelevation observations, we incorporate hints from road design principles as auxiliary information to improve the robustness of bias/offset estimation. Next, we present the proposed methodology to estimate the offsets.

3.4.1 Identification of Horizontal Curves To identify horizontal curves on a route, we filter out sensor traces based on yaw rate of vehicle ($\omega_{Z_c,t}$). Specifically, we use $turn_{thresh}$ and keep the road sections where yaw rate > $turn_{thresh}$. To avoid false positives due to turns (for e.g. at intersections) and lane changing events, we keep road sections with length > 100 m. We represent the identified horizontal curves on a route by set H. Moreover, we filter out the straight sections of a road segment. We select sections of length >50 m, where yaw rate < $turn_{thresh}$. Empirically driven value of $0.01rad/sec$ is used as $turn_{thresh}$.

3.4.2 Vehicle Kinematics Model Fig. 6 illustrates the kinematics of a vehicle negotiating a circular curve with radius R and superelevation e. Balancing out forces along X_c and solving for e, we get Eq. 7 where, V is velocity of the vehicle and g is the acceleration due to gravity ($9.81\ m/sec^2$). We assume θ to be 0 (tilt of the vehicle w.r.t the surface of the road), which is a reasonable assumption to make for modern vehicles characterized by firmer suspension systems as compared to their older counterparts. Information needed to estimate superelevation i.e. the radius of the curve, lateral acceleration and velocity can be sensed by an on-board smartphone. The velocity of the vehicle and radius of the curve are acquired using GPS sensor of the smartphone. Next, we describe the proposed methodology to estimate radius profile of a curve.

$$e = \arccos\left(\frac{A_{X_c}}{\sqrt{\left(\frac{V^2}{R}\right)^2 + g^2}}\right) - \arccos\left(\frac{\left(\frac{V^2}{R}\right)^2}{\sqrt{\left(\frac{V^2}{R}\right)^2 + g^2}}\right) \tag{7}$$

3.4.3 Radius Estimation GPS traces corresponding to identified horizontal curves in set H are first map matched to the centerline

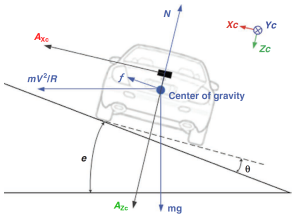

Figure 6: Vehicle kinematics on a curve.

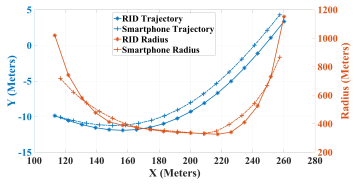

Figure 7: Curve trajectory and radius estimation.

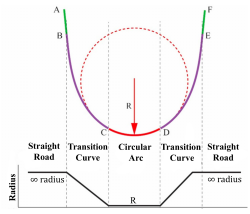

Figure 8: Horizontal curve design.

of a road segment using Google's Map Matching API [6]. Map matching filters out noisy estimations of vehicle's position from smartphone's GPS. We convert the GPS latitude and longitude to earth-centered, earth-fixed (ECEF) cartesian coordinates [26]. The cartesian coordinates are interpolated using distance based interpolation, to get equally spaced points along the curve [12]. The curvature profile is estimated using [1], where radius at a point P_K is approximated by the radius of a circle passing through it and its neighboring points P_{K-1} and P_{K+1}. Fig. 7 illustrates the estimated points and radius profile along a horizontal curve using our proposed approach along with the ground truth.

3.4.4 Superelevation Estimation As described in Sec. 3.4.2, Eq. 7 holds for circular curves. Therefore, there is a requirement to filter out sensor traces on the circular portions of horizontal curves to estimate superelevation using the model. We leverage insights on horizontal curve design principles to extract the corresponding sensor traces on the circular portion.

To ensure driver comfort and safety, horizontal curve design is typically done using two transition curves and a circular arc. To avoid sudden changes in lateral acceleration, transition curves are used between the straight sections and the circular arc, allowing smooth variations of the lateral forces on the vehicle. The horizontal curve design is illustrated in Fig. 8, where the straight sections AB and EF of the road are transitioned to the circular arc CD using two transition curves, BC and DE. Moreover, the transverse slope angle (or cross slope) on the straight road section is gradually changed to the superelevation angle in the circular arc section (Fig. 1).

The manifestation of the above design principles can be seen in Fig. 9, where the relationship between ground truth radius profile of a sample horizontal curve with the sensed transverse slope, lateral acceleration and yaw rate from smartphone is shown. The corresponding straight, transition and circular sections from Fig. 8 are marked for reference. The radius profile shows the transition sections "BC" and "DE", where the curvature is gradually changed to the constant curvature (or radius) in the circular section "CD". The transverse slope profile also increases and decreases gradually in "BC" and "DE", respectively. The superelevation angle is attained the circular portion of the horizontal curve "CD".

We leverage the transverse slope profile (derived using method described in Sec. 3.3) to filter out circular sections from the horizontal curves in set H. Since, transverse slope attains maximum/minimum value (the superelevation) in the circular section of a horizontal curve, we extract the global maxima or minima (sup_{max}) of the transverse slope profile and search for the largest set of contiguous points $S = \{s_{x1}, s_{x2}, s_{x3}, \ldots, s_{xn}\}$, whose values are in the range $sup_{max} - 0.5 < sup_{max} < sup_{max} + 0.5$. To account for noisy data from smartphones, we sample points every 10 meters from the transverse slope profile to get S. This extracted region is denoted by $[x_{strt}, x_{end}]$, where x_{strt} and x_{end} are positions of the start s_{x1} and the end s_{xn} of the circular section, respectively. Mean of the radius, lateral acceleration and velocity profiles in the circular section are used as the input to Eq. 7 to estimate superelevation of a particular horizontal curve.

Fig. 10 compares the estimation of radius and superelevation of horizontal curves on the test route estimated using methodology described above with the ground-truth ("RID"). Mean of superelevations derived from 15 trips is shown in the figure. The estimations derived from the vehicle dynamics model can be unreliable (e.g., Curve ID 3 and 5 in Fig. 10). In particular, we identify the following causes for unreliable estimation of superelevation using the vehicle dynamics model : **a) GPS noise :** Velocity data from GPS can be noisy, especially in urban environments, where occlusions due to surrounding buildings, foliage, etc. result in weak signal strength [7]. Furthermore, errors in superelevation estimation can arise from inaccurate estimation of radius due to low resolution data from digital map data [15]. **b) Susceptibility to driving maneuvers :** Lateral lane keeping maneuvers performed by the driver while negotiating a curve will result in deviation of the vehicle from following a perfectly circular trajectory. These maneuvers result in significant variations in the lateral acceleration sensed by the smartphone and thus influence superelevation estimation. **c) Simple Model :** The kinematics model used is fairly simple (assumes vehicle to be a point mass) and fails to capture detailed dynamics of the vehicle for more accurate estimation of superelevation. Approaches using complex vehicle dynamics models [9] that take into account vehicle properties such as chassis dimensions, suspension properties, etc. are not suitable for our application scenario, that solely relies on smartphone data for road transverse profile estimation.

Due to the aforementioned factors, estimations of superelevation from vehicle kinematics model are noisy and can be unreliable "fixes" to correct the offset. To handle this, we frame an optimization problem that accounts for noisy superelevation estimations from vehicle kinematics model by incorporating information from road design principles. We describe the proposed method in the next section.

3.4.5 Offset Estimation As described in the previous section, solely relying on superelevation estimation from vehicle kinematics model to estimate the offsets can be unreliable. To solve the above problem, we design an optimization framework to estimate the offsets associated with transverse slope profile estimation. The framework

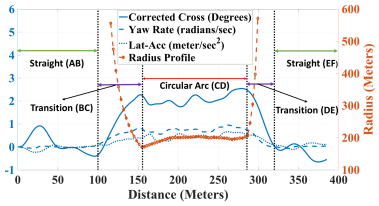

Figure 9: Response of smartphone sensors while negotiating a curve.

Curve ID	Radius (Meters)		Superelevation (Degrees)	
	Smartphone	RID	Model	RID
1	240	263	-4.05	-3.9
2	152	150	3.6	3.8
3	185	226	-4.6	-3.5
4	177	171	4.11	4.5
5	295	335	-3.15	-2.2

Figure 10: Superelevation estimation.

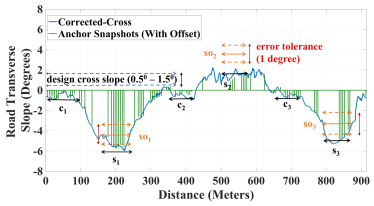

Figure 11: Illustration of the offset estimation methodology.

takes hints from road design principles and incorporates uncertainty in estimation of superelevation. The design of the proposed framework is based on the following observations : **a)** Although error prone, observations of super elevation derived from the model can provide a reasonable hypotheses on the "true" superelevation of a curve. **b)** Generally, the cross slope ranges between **1.5% - 2% (0.86° - 1.12°)** as prescribed by the road design principles [4]. The role of the cross slope in road design is to facilitate drainage of rainwater. Cross slopes that are too steep can cause vehicles to drift and skid laterally when braking. Therefore, both the minimum and the maximum values of cross slope are important criteria for optimal road design.

Using the above observations as constraints, we formulate a linear program to estimate the offsets. For each $i, j \in \{1, \cdots, h\}$, where i and j are indices of a pair of horizontal curves from set H containing h horizontal curves on the route, we have the following formulation :

$$\begin{aligned} \text{minimize} \quad & |o_i - o_j| \\ \text{subject to} \quad & 0.5 < c_i + o_i < 1.5 && (a) \\ & 0.5 < c_j + o_j < 1.5 && (b) \\ & so_i - 0.5 < s_i + o_i < so_i + 0.5 && (c) \\ & so_j - 0.5 < s_j + o_j < so_j + 0.5 && (d) \end{aligned}$$

Fig. 11 illustrates the decision variables and parameters used in the optimisation model. The decision variables are o_i and o_j which are the offsets. Following are the parameters of the model. s_i and s_j are the observations of superelevation from the biased estimation of transverse slope profile for i^{th} and j^{th} curve, respectively. c_i and c_j are the observation of cross slopes from the biased estimation of transverse slopes profile of i^{th} and j^{th} curves, respectively. so_i and so_j are the observations of superelevation made from the vehicle kinematics model for i^{th} and j^{th} curves, respectively. Mean estimations of superelevation from all the trips are used as so_i and so_j. s_i, s_j, c_i and c_j are estimated using the mean of the observations from the transverse slope profile (from an individual trip) in the circular and straight sections of a road segment. The straight and circular sections are extracted using the methodology described in Sec. 3.4.1.

The proposed framework searches for an optimal alignment (or offsets) of the biased transverse slope profile in a space constrained by **a)** superelevation observations from vehicle kinematics model **b)** cross slope design criteria. As explained in Sec. 3.4, the offsets arise from the errors in the coordinate alignment procedure. Therefore, the offsets should be same through the course of a trip as long as the smartphone's orientation w.r.t vehicle does not change. We achieve this by minimising the objective function defined as the absolute difference between offsets on a pair of horizontal curves. In constraints $(a) - (b)$, we incorporate the road design criteria for cross slope (0.5° - 1.5°). Furthermore, as observed in Sec. 3.4.4, the estimations of superelevation using vehicle kinematics model can be unreliable. Therefore, we incorporate an error tolerance of 1° on the estimations of superelevation in constraints $(c) - (d)$.

The above mathematical program is used to model all the pairs of horizontal curves in the set H to generate a set of offset pairs (o_i, o_j), for all $i, j \in \{1, ..., h\}$. The mean of a o_i-o_j pair is considered as the offset estimation for that particular pair of curves. Finally, the mean of offsets of all the pairs of curves from the set H is used to correct the offset/bias associated with anchor snapshots estimation from an individual trip (Fig. 2, left).

3.5 Aggregation Framework

Due to its ubiquity and low-cost, smartphone as a sensing platform is an attractive option to develop crowd-sourced frameworks, to perform large scale sensing tasks. In our application, we can take advantage of observations of road transverse slope from multiple sources to increase accuracy and robustness of our system. In a typical crowd-sourced system, sources can provide conflicting observations on an object due to varying QoI (Quality of Information) from multiple sources. Specific to our application, varying QoI of sources arises from factors such as varying a) quality of sensors in smartphones, b) suspension properties of different vehicles, c) inherent vibration of different vehicles, d) quality of phone holders used to mount the smartphone, etc. As an example, accelerometer and gyroscope estimations from a phone "loosely" mounted in the vehicle will experience more noise than that of a firmly mounted phone, especially on roads in poor condition.

To handle the problem of varying QoI of sources, truth discovery methods [13, 14] are proposed, which take into account source reliability into data aggregation. Sources are assigned weights based on their QoI i.e. reliable sources with high QoI are given more weight and vice-versa. We leverage CRH [13] to perform aggregation of observations from different sources. CRH formulates the observation conflicts from different sources as an optimization problem to minimize the overall weighted distance between the input and the estimated truths.

The aggregation of estimations from various trips is done in two steps a) *Aggregation of Anchor Snapshots*, and b) *Profile Aggregation*.

3.5.1 Aggregation of Anchor Snapshots As described in Sec. 3.3.1, we opportunistically filter out "anchor snapshots" from estimates of vehicle's roll using accelerometer when the vehicle dynamics are stable. Due to this the estimations of anchor snapshots can be sparse.

The "density" and location of anchor snapshots on road segment is dependent on occurrence of stable driving events, thus is influenced by factors such as driving behavior of the user, traffic conditions, etc. For example, frequent rapid acceleration and deceleration or lane changing events, due to the user being aggressive or traffic congestion will result in fewer observations of anchor snapshots during a trip. Fewer observations mean less information for drift correction to work with. To handle the sparsity, we first aggregate estimations of anchor snapshots (after offset/bias correction) from various trips on a given road segment (Fig. 2, top-left). The intuition is to handle the sparse estimations of anchor snapshots described above by increasing the density of anchor snapshots on a given road segment, using data from different trips. Furthermore, aggregation will compensate the effects of varying QoI from different sources. We divide the road segment into bins of length bin_{acc} (set to 2m) and sample the observations from different trips in these bins. Observations in each bin are then aggregated to produce the final output.

3.5.2 Profile Aggregation We apply the drift correction method described in Sec. 3.3.2 on roll estimations from gyroscope of different trips using the aggregated anchor snapshots on a given road segment (Fig. 2, top). As done in Sec. 3.5.1, we divide the road segment into bins of length $bin_{corr-gyr}$. Compared to bin_{acc}, $bin_{corr-gyr}$ is smaller (set to 20cm) due to presence of continuous observations from gyroscope. The corrected gyroscope estimations thus derived are finally aggregated into a single profile of road cross slope on a given road segment. The intuition is to "average out" the discrepancies in estimations using gyroscope due to factors such as quality of sensors on the phone, quality of phone holder, etc.

4 Evaluation

4.1 Experimental Setup

Data. Data collection was done on a test route shown in Fig. 12 using 3 different smartphones: Nexus 5, Nexus5x and Google Pixel XL. We leverage VehSense, an android application, that was developed for data collection. Smartphones were fixed in arbitrary orientation in the vehicle using phone-holders at various locations such as wind-shield, air-conditioning vents, etc. Specifically we collect the following time-series samples: a) 3-axis angular velocity data from the gyroscope, b) 3-axis acceleration data from the accelerometer, and c) GPS data including latitude, longitude, speed and bearing.

Data collection was done over the course of one month in May 2019 in diverse traffic scenarios, ranging from high traffic during peak hours (≈ 9am-10am and ≈ 5pm-6pm) to low traffic (≈ 3pm-4pm and ≈ 10pm - 11pm). The collected data is comprised of 15 trips distributed over 5 participants each having a different vehicle. For coordinate alignment, each participant was asked to keep the vehicle stationary for ≈ 30 seconds in the parking lot, before starting the trip at Point A on the map (Fig. 12). The participants were asked to drive naturally on the route, which is ≈ 9 km.

Comparison. We evaluate the performance of our proposed method by comparing it to [20] (labeled "Baseline" in rest of the paper), which uses a mobile device to estimate road transverse slope profile. It uses a complimentary filter to fuse observations from gyroscope and accelerometer, the key idea of which is to combine

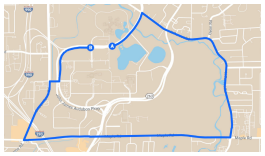

Figure 12: Test Route.

precise short term estimations from gyroscope with the long term accumulated estimations from accelerometer.

Groundtruth. For groundtruth, we use data from the Road Inventory Database [11], which includes information on road geometry features (curvature, grade, cross slope and superelevation) of ~25,000 directional miles of roadway in six sites in USA. The data is collected using ARAN (Automatic Road Analyzer)[5], a specialized instrumented vehicle with high-grade IMU's, laser scanners, high-precision GPS, and camera.

Evaluation Measure: We use the following to compare the performance of our system with the baselines:

- **Absolute Error (AE):** It is the absolute value of difference between our estimation and the ground-truth at a point on the road.
- **Gradient Error (GE):** It is the absolute value of difference of change of road transverse slope per unit distance between our estimation and the ground-truth. This metric will indicate the performance of various methodologies in capturing the shape of the road profile.

We list the abbreviations used in the paper:

- **Indiv-Anch:** Anchor snapshots estimated using method in Sec. 3.3.1 and corrected by applying offsets using method in Sec. 3.4.
- **Indiv-Prof:** Transverse slope profiles generated using Indiv-Anch by applying drift correction using method in Sec. 3.3.2 (i.e. without applying data aggregation method in Sec. 3.5).
- **Agg-Anch:** Aggregated anchor snapshots generated by aggregating Indiv-Anch using method in Sec. 3.5.1.
- **Agg-Anch-Prof:** Transverse slope profiles profiles generated using Agg-Anch by applying drift correction using method in Sec. 3.3.2.
- **Agg-Prof-Final:** Aggregated grade profiles generated by aggregating Agg-Anch-Prof using method in Sec. 3.5.2.
- **Indiv-Anch-No-Off:** Anchor snapshots estimated using method in Sec. 3.3.1, but without applying offsets from Sec. 3.4.
- **Agg-Anch-No-Off:** Aggregated anchor snapshots generated by aggregating Indiv-Anch-No-Off using method in Sec. 3.5.1.
- **Agg-Anch-Prof-No-Off:** Transverse slope profiles profiles generated using Agg-Anch-No-Off by applying drift correction using method in Sec. 3.3.2.
- **Agg-Prof-Final-No-Off:** Aggregated transverse slope profiles generated by aggregating Agg-Anch-Prof-No-Off using method in Sec. 3.5.2.

4.2 Overall Performance

In this section, we analyze the overall performance of our system and compare it with the baseline. Fig. 13, 14, and 15 show the comparison of road transverse slope profiles estimated using our

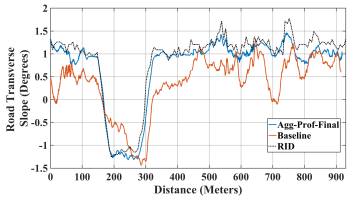

Figure 13: Road segment 1.

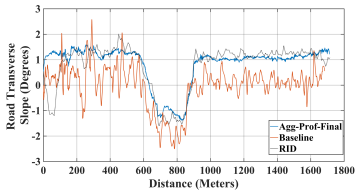

Figure 14: Road Segment 2.

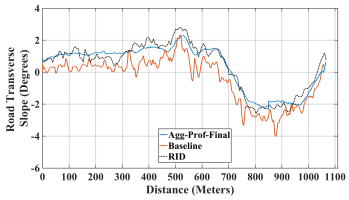

Figure 15: Road Segment 3.

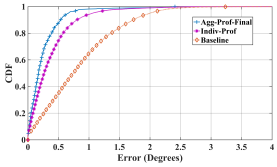

Figure 16: Absolute Error

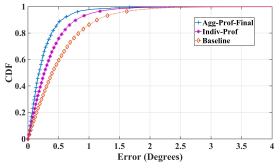

Figure 17: Gradient Error

approach and the baseline on different road segments. The figures demonstrate the effectiveness of our proposed approach (labeled "Agg-Prof-Final") to capture the transverse slope profile of road segments. On the other hand transverse slope profiles generated by the baseline are error prone with high variance.

Fig. 16 shows the statistical comparison of absolute error (AE) of different approaches on the entire test route. Our proposed approach outperforms the baseline. The 50% and 90% absolute error (AE) of road transverse slope profile estimated using the baseline are 0.73° and 1.7°, respectively. In comparison, our approach without aggregation (labeled "Indiv-Prof") has 50% and 90% AE of 0.27° and 0.8°, respectively.

Gains in accuracy of our approach over transverse slope estimation using the baseline is also observed in Fig. 17, which illustrates gradient error (GE) of different approaches on the entire test route. The baseline's 50% and 90% GE are 0.4° and 1.1°, respectively. In comparison, our approach without aggregation (labeled "Indiv-Prof") has 50% and 90% GE of 0.26° and 0.79°, respectively.

Our approach without applying data aggregation (labeled "Indiv-Prof") outperforms the baseline in both Absolute Error and Gradient Error, demonstrating the impact of intelligently combining the estimations from accelerometer and gyroscope based on understanding of their error characteristics in a driving environment. In particular, the baseline suffers in accuracy mainly due to susceptibility to vehicle dynamics. To exploit the complimentary nature of sensors, the baseline adds/fuses the low-frequency component of accelerometer estimation with high-frequency component of gyroscope's estimate to estimate the tilt of the vehicle w.r.t the ground. The estimated tilt is added to the superelevation derived from the vehicle dynamics model. However, optimal parameter setting (e.g., cut-off frequency for the filters) is difficult to achieve because of varying road types and dynamic noise profiles of smartphone's sensors. This makes the baseline susceptible to both absolute and gradient errors, especially due to pollution of estimation from accelerometer during periods when the dynamics of vehicle are not stable. For e.g. in Fig. 13, the peak at $\approx 710 meters$ is due to lateral lane-keeping maneuver performed by the driver. Similarly, in Fig. 14 the peaks at $\approx 250 - 300 meters$ are due to lane changing events performed by the driver. Our approach, on the other hand, relies primarily on estimation from gyroscope which is not prone to errors induced by vehicle dynamics.

Next we analyse the impact of data aggregation. Gains in accuracy due to aggregation of estimations from various trips is evident from the results as shown in Fig .16 and 17. The 50% and 90% AE of aggregated road cross slope estimations (labeled "Agg-Prof-Final") are 0.17° and 0.53°, respectively. The 50% and 90% GE are 0.18° and 0.56°, respectively. The results indicate the power of data aggregation, which essentially "averages out" errors in estimation due to varied data quality of smartphone sensors.

4.3 Impact of Offset Correction

Fig. 18 illustrates the impact of offset correction methodology described in Sec. 3.4. Offset correction in general result in accuracy gains. The 50% and 90% gains in accuracy of "Indiv-Prof" over "Indiv-Prof-UnOff" are 0.5° and 1.9°, respectively. The 50% and 90% gains in accuracy of "Agg-Prof" over "Agg-Prof-UnOff" are 0.6° and 0.68°, respectively.

A marginal improvement of "Indiv-Prof" over "Indiv-Prof-Naive" illustrates the increase in robustness of the system due incorporation of information on cross slope design principles in the methodology described in Sec 3.4.5. The offsets for "Indiv-Prof-Naive" were estimated by simply using $o_i = so_i - s_i$, where i=1 ,..., c. Mean of "o_i"'s was used as final offset to estimate "Indiv-Prof-Naive".

4.4 Data Aggregation

Fig. 19 illustrates the impact of performing aggregation on estimates of anchor snapshots. In general, aggregated estimations of anchor snapshots result in accuracy gains. The 50% and 90% gain in accuracy of aggregated anchor snapshots (labeled "Agg-Anch-CRH") over "Indiv-Anch" is 0.1° and 0.15°, respectively. Furthermore, the performance of aggregated estimation computed using CRH is better than simply averaging the anchor snapshots (labeled "Agg-Anch-AVG").

Better performance of "CRH" over "Average" can also be seen in Fig. 20, which shows the AE distributions for the estimated road transverse slope profiles. CRH consistently performs better than average as it takes into account source reliability by assigning more weight to reliable sources.

4.5 Superelevation Estimation

Table. 1 illustrates the performance comparison of superelevation estimations using vehicle kinematics model described in Sec. 3.4.2 and using our method (labeled "Ours"). Mean of estimation from 15 trips was used to derive the final superelevation of a horizontal curve (labeled "Kinematics Model"). "Ours" is estimated by averaging the estimations from "Agg-Prof-Final" in the circular section of the horizontal curve (Sec. 3.4.4). The average absolute error for

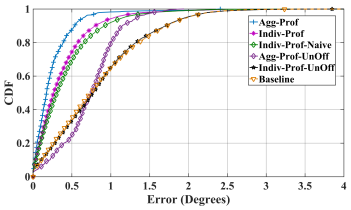

Figure 18: Impact of offset correction.

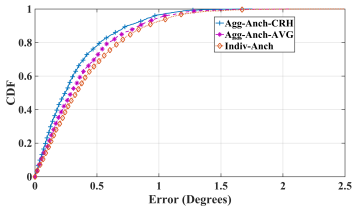

Figure 19: Anchor snapshots aggregation.

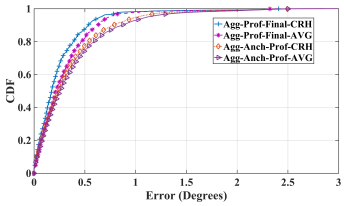

Figure 20: Profile Aggregation.

Curve-ID	Ground Truth Superelevation	Kinematics Model	Absolute Error (Model)	Ours	Absolute Error (Ours)
1	-1.12	-1.9	0.78	-1.04	0.08
2	-2.2	-2.1	0.1	-1.92	0.3
3	2.6	1.43	1.17	2.26	0.34
4	-1.3	-0.8	0.5	-1.5	0.2
5	-1.09	-0.7	0.4	-1.1	0.01
6	-1.57	-1.5	0.07	-1.42	0.15
7	-3.8	-4	0.2	-3.6	0.2
8	4	3.6	0.4	3.6	0.4
9	-3.34	-4.6	1.26	-3.2	0.14
10	3.51	4.11	0.5	3.2	0.3
11	-1.33	-1.65	0.32	-1.1	0.23

Table 1: Comparison of superelevation estimations using vehicle kinematics model and our approach.

11 curves using the kinematics model and our method are 0.52° and 0.23°, respectively. The variance of errors in estimations of superelevation using the model and our approach are 0.16 and 0.013, respectively. This illustrates the efficacy of our proposed system to estimate superelevation of horizontal curves both accurately and reliably.

5 Conclusion

This paper presents a novel, cost-efficient and easily deployable crowd-sourcing system for road transverse slope estimation using smartphones. Deriving insights from analysis of smartphone sensor's error characteristics in a dynamic driving environment, we intelligently integrate data from accelerometer, gyroscope and observations from vehicle kinematics model to estimate road transverse slope. Finally, we crowd-source observations from various sources to improve accuracy and robustness of the system. The experiment results demonstrate that the proposed method provides considerable improvement over existing solutions.

References

[1] Are Mjaavatten (2020). [n.d.]. Curvature of a 2D or 3D curve. https://www.mathworks.com/matlabcentral/fileexchange/69452-curvature-of-a-2d-or-3d-curve [Online; accessed 1-JUN-2020].

[2] T Baybura and O Baykal. 2001. The Investigation Of The Effect Of The Vertical Alignment Geometry On The Lateral Change Of Acceleration. *Istanbul Technical University, Istanbul, Turkey* (2001).

[3] Adam J Dean and Sean N Brennan. 2009. Terrain-based road vehicle localization on multi-lane highways. In *2009 American Control Conference*. IEEE, 707–712.

[4] Federal Highway Administration. [n.d.]. Mitigation Strategies For Design Exceptions. https://safety.fhwa.dot.gov/geometric/pubs/mitigationstrategies/chapter3/3_crossslope.cfm [Online; accessed 1-JUN-2020].

[5] FUGRO. [n.d.]. ARAN - Automatic Road Analyzer. https://www.fugro.com/our-services/asset-integrity/roadware/aran-automatic-road-analyzer [Online; accessed 7-March-2019].

[6] Google. [n.d.]. Snap To Roads. https://developers.google.com/maps/documentation/roads/snap [Online; accessed 1-JUN-2020].

[7] Paul D Groves, Lei Wang, and M Ziebart. 2012. Shadow matching: Improved GNSS accuracy in urban canyons. *GPS World* 23, 2 (2012), 14–18.

[8] Abhishek Gupta, Shaohan Hu, Weida Zhong, Adel Sadek, Lu Su, and Chunming Qiao. 2020. Road Grade Estimation Using Crowd-Sourced Smartphone Data. In *2020 19th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*. IEEE, 313–324.

[9] Ehsan Hashemi, Reza Zarringhalam, Amir Khajepour, William Melek, Alireza Kasaiezadeh, and Shih-Ken Chen. 2017. Real-time estimation of the road bank and grade angles with unknown input observers. *Vehicle system dynamics* 55, 5 (2017), 648–667.

[10] HERE. [n.d.]. HD-MAPS. https://www.here.com/products/automotive/hd-maps [Online; accessed 7-March-2019].

[11] Iowa State University. [n.d.]. SHRP2-RID. https://ctre.iastate.edu/shrp2-rid/ [Online; accessed 7-March-2019].

[12] John D'Errico (2020). [n.d.]. Interparc-Distance based interpolation along a general curve in space. https://www.mathworks.com/matlabcentral/fileexchange/34874-interparc [Online; accessed 1-JUN-2020].

[13] Qi Li, Yaliang Li, Jing Gao, Bo Zhao, Wei Fan, and Jiawei Han. 2014. Resolving conflicts in heterogeneous data by truth discovery and source reliability estimation. In *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*. ACM, 1187–1198.

[14] Yaliang Li, Jing Gao, Chuishi Meng, Qi Li, Lu Su, Bo Zhao, Wei Fan, and Jiawei Han. 2016. A survey on truth discovery. *ACM Sigkdd Explorations Newsletter* 17, 2 (2016), 1–16.

[15] Zhixia Li, Madhav V Chitturi, Andrea R Bill, and David A Noyce. 2012. Automated identification and extraction of horizontal curve information from geographic information system roadway maps. *Transportation research record* 2291, 1 (2012), 80–92.

[16] Mohammad Mahanpoor, Saeed Monajjem, and Vahid Balali. 2019. Sustainable highway maintenance: optimization of existing highway vertical alignment considering pavement condition. *Sustainability* 11, 6 (2019), 1659.

[17] Alexander Mraz and Abdenour Nazef. 2008. Innovative techniques with a multipurpose survey vehicle for automated analysis of cross-slope data. *Transportation Research Record* 2068, 1 (2008), 32–38.

[18] Myung-Wook Park, Sang-Woo Lee, and Woo-Yong Han. 2014. Development of lateral control system for autonomous vehicle based on adaptive pure pursuit algorithm. In *2014 14th International Conference on Control, Automation and Systems (ICCAS 2014)*. IEEE, 1443–1447.

[19] Reginald Souleyrette, S Hallmark, Sitansu Pattnaik, Molly O'Brien, and David Veneziano. 2003. *Grade and cross slope estimation from LiDAR-based surface models*. Technical Report.

[20] Yichang Tsai and Chengbo Ai. 2017. Automated Superelevation Measurement Method Using a Low-Cost Mobile Device: An Efficient, Cost-Effective Approach Toward Intelligent Horizontal Curve Safety Assessment. *Transportation Research Record* 2621, 1 (2017), 62–70.

[21] Yichang Tsai, Chengbo Ai, Zhaohua Wang, and Eric Pitts. 2013. Mobile cross-slope measurement method using lidar technology. *Transportation research record* 2367, 1 (2013), 53–59.

[22] UMICH. [n.d.]. Curve Speed Warning Systems. http://www.umtri.umich.edu/our-focus/curve-speed-warning-systems [Online; accessed 7-March-2019].

[23] US Department of Transportation. [n.d.]. Estimated Lane-length (1980 - 2017) statistics. https://www.fhwa.dot.gov/policyinformation/statistics/2017/hm260.cfm [Online; accessed 1-FEB-2020].

[24] Yan Wang, Jie Yang, Hongbo Liu, Yingying Chen, Marco Gruteser, and Richard P Martin. 2013. Sensing vehicle dynamics for determining driver phone use. In *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*. ACM, 41–54.

[25] Wikipedia. [n.d.]. Cross Slope. https://en.wikipedia.org/wiki/Cross_slope [Online; accessed 1-JUN-2020].

[26] Wikipedia. [n.d.]. ECEF. https://en.wikipedia.org/wiki/ECEF [Online; accessed 1-JUN-2020].

[27] Wikipedia contributors. [n.d.]. Advanced driver-assistance systems. https://en.wikipedia.org/wiki/Advanced_driver-assistance_systems [Online; accessed 7-March-2019].

[28] Jonathan S Wood and Shaohu Zhang. 2018. Identification and calculation of horizontal curves for low-volume roadways using smartphone sensors. *Transportation research record* 2672, 39 (2018), 1–10.

[29] Derek York. 1966. Least-squares fitting of a straight line. *Canadian Journal of Physics* 44, 5 (1966), 1079–1086.

[30] Pengfei Zhou, Mo Li, and Guobin Shen. 2014. Use it free: Instantly knowing your phone attitude. In *Proceedings of the 20th annual international conference on Mobile computing and networking*. ACM, 605–616.