Inferring high-resolution traffic accident risk maps based on satellite imagery and GPS trajectories

Songtao He
MIT CSAIL
songtao@mit.edu

Mohammad Amin Sadeghi
HBKU QCRI
MSadeghi@hbku.edu.qa

Sanjay Chawla
HBKU QCRI
schawla@hbku.edu.qa

Mohammad Alizadeh
MIT CSAIL
alizadeh@csail.mit.edu

Hari Balakrishnan
MIT CSAIL
hari@csail.mit.edu

Samuel Madden
MIT CSAIL
madden@csail.mit.edu

Abstract

Traffic accidents cost about 3% of the world’s GDP and are the leading cause of death in children and young adults. Accident risk maps are useful tools to monitor and mitigate accident risk. We present a technique to generate high-resolution (5 meters) accident risk maps. At this high resolution, accidents are sparse and risk estimation is limited by bias-variance trade-off. Prior accident risk maps either estimate low-resolution maps that are of low utility (high bias), or they use frequency-based estimation techniques that inaccurately predict where accidents actually happen (high variance). To improve this trade-off, we use an end-to-end deep architecture that can input satellite imagery, GPS trajectories, and road maps and the history of accidents. Our evaluation on four metropolitan areas in the US with a total area of 7,488 km² shows that our technique outperform prior work in terms of resolution and accuracy.

1. Introduction

According to WHO, each year 1.35 million people die and 20 to 50 million people sustain non-fatal injuries from traffic accidents [20]. In the US alone, traffic accidents cost $871 billion annually [6]. In most countries traffic accidents cost about 3% of the GDP [20]. By identifying high-risk locations on the map, many groups, including drivers, police departments, transportation departments and insurance companies can take actions to reduce this risk.

Accident risk maps assign an expected rate of accident over a given time period to each location on the map. Prior works predict accident maps with resolutions of a few hundred meters (Table 1). In this work we predict maps with 5m×5m resolution because there are important details that are not captured in lower resolutions. At this resolution, sparsity causes a bias-variance trade-off in the estimation of the underlying risk. We explain this challenge in Section 2.

We improve this trade-off by incorporating context information from satellite imagery, GPS trajectories, and road maps. It outputs accident probability distribution. Note that our model has identified a few locations as high-risk (highlighted with circles) even though they have no historical accidents. Locations that our model has identified as high-risk experienced accidents during the follow-up years.

Figure 1. Our model inputs road maps, satellite imagery, GPS trajectories, and historical traffic accidents. It outputs accident probability distribution. Note that our model has identified a few locations as high-risk (highlighted with circles) even though they have no historical accidents. Locations that our model has identified as high-risk experienced accidents during the follow-up years.

At 5m×5m resolution, evaluation is also challenging because the ground-truth is noisy (It is sampled from a hidden risk distribution.) In Section 4 we present a process to estimate the prediction error with respect to the true underlying risk distribution. Our maps outperform prior work in terms of resolution and prediction error.
2. Challenge of sparsity

In the US, the average annual rate of reported accidents on a 5m×5m block of road is about 1 in 1000. Our analysis on US traffic accidents dataset [16] shows that 31% of the accidents occur in places where no other accidents happened nearby (within 50 meters) within four years. Therefore, Monte Carlo probability estimation will miss some high-risk areas and misidentify low-risk areas as high-risk.

Our goal in accident risk prediction is not to identify exactly where new accidents will occur because this is impossible. Instead, our goal is to identify the underlying risk of accidents at each location, whether accidents occur or not. Ideally, we should use the underlying risk of accidents as ground-truth. However, the underlying risk of accidents is unknown. Therefore, we use a map of future accidents as an alternative ground-truth. The map of the future accidents is a Monte Carlo estimation of the underlying rate of accidents, so it carries a large amount of estimation error. This error leads to challenges in both prediction and evaluation.

Prediction: Assume that a 5m×5m grid cell has a 1% annual rate of accidents. In one year, the number of accidents at this location would be either 0 or 1. Estimating the true risk of 1% given an input of 0 or 1 is challenging.

Evaluation: Since the true underlying rate of accidents is unknown, we can only use an observed rate of accident as a proxy to ground-truth. Therefore, our ground-truth itself carries error and this adversely affects the evaluation.

One way to deal with the challenges of sparsity is to reduce estimation resolution. However, this low resolution causes bias in estimation. As an example of this bias, assume that a dangerous intersection and a safe street are grouped into one single cell. An estimate for the risk in this cell will underestimate the risk of the dangerous intersection and will overestimate the risk of the safe street (Figure 2-c). This underestimation and overestimation repeats with varying input, therefore, it is a form of model bias.

2.1. Prior work

Most prior works use low resolutions to control sparsity (Table 1). Several works in transportation journals adopt a high resolution, but they are used for visualization purposes and do not evaluate their results.

The general problem of accident prediction is a frequent subject of study [27], but few studies produce accident maps. Most works focus on the effect of certain events (weather, calendar, lighting) on the frequency of accidents [13]. In this work, we study the spatial accident maps and only review major prior works that produce accident maps. These works either produce coarse resolution maps or use kernel density estimation.

Coarse resolution: Some works choose a coarse spatial granularity to predict accident risk. The most relevant works include the work by Najjar et al. [17] that uses satellite imagery, and the work by Chen et al. [11] that uses GPS trajectories. Other notable works are listed in Table 1. Coarse-resolution models miss accident hot-spots and misidentify low risk locations as high risk (Figure 2-c).

Kernel Density Estimation: Most works in transportation journals use Kernel Density Estimation. KDE applies a Gaussian kernel to historical measurements [23]. Xie et al. [25] use KDE along the roads rather than in the 2D domain. Most KDE works evaluate their performance only visually. The most notable exception is the work by Xie et al. [26] that calculates statistical significance levels. Anderson et al. [2] identify accident hot-spots but do not quantitatively evaluate the performance. Le et al. [14] identify hot-spots and evaluate their ranking. All of these KDE-based works only use historical accidents and a road map to visualize accidents [21, 5, 24, 3, 31]. These works use similar KDE techniques in different cities around the world. Okabe et al. [19], Netek et al. [18] and Shariat et al. [15] implemented KDE in GIS environment. We implemented KDE and compared against it (Figure 2).
are many pedestrians. lanes, whether there is a road shoulder, and whether there
the density, speed, and flow of traffic. Satellite imagery
information. For example, GPS trajectories carry information about
historical accident data) provide complementary informa-
on an accident-based similarity score.
for each position. Our model learns an internal metric based
trajectories, and road maps. We use a deep model that in-
ity between intersections.
need some context information that can capture the similar-
lar accident risk profiles. If one intersection experiences as
appearances, and similar traffic patterns are likely to have sim-
that places with similar road structures, similar visual ap-
Monte Carlo estimation. As such, it only works for places
experience accidents in the future. Note that the KDE predicts very
specific risky locations, many of which do not have accidents in
traffic. In contrast, our method accurately highlights the roads
where future accidents happen, properly attributing more risk to
intersections and ramps. We did not visualize historical accidents
as they look similar to KDE.

2.2. Addressing the challenge of sparsity

Prior work based on historical accidents uses variants of
Monte Carlo estimation. As such, it only works for places
where there is sufficient historical accident data and well-
maintained records. To overcome this challenge, we note
that places with similar road structures, similar visual ap-
pearances, and similar traffic patterns are likely to have sim-
lar accident risk profiles. If one intersection experiences as
accident, we can share some risks with similar intersections.
In order to generalize from one intersection to another, we
need some context information that can capture the similarity
between intersections.

In this work, we use context from satellite imagery, GPS
trajectories, and road maps. We use a deep model that in-
puts context and learns to generate useful representations
for each position. Our model learns an internal metric based
on an accident-based similarity score.

The three data modalities that we used (in addition to
historical accident data) provide complementary informa-
. For example, GPS trajectories carry information about
the density, speed, and flow of traffic. Satellite imagery
carries information about the road, such as the number of
lanes, whether there is a road shoulder, and whether there
are many pedestrians.

2.3. Evaluation with ground-truth error

We use future accidents as a proxy to the underlying rate
of accidents. This proxy has an error that adversely affects
evaluation, therefore we need to isolate it.

Assume a map has \( n \) grid cells and there is an underlying
rate of accident for each cell \( R_i \) that we want to estimate.
There is an observation of the historical rate of accidents
at each location \( H_i \). There is also an observation of the
future rate of accident \( F_i \). Since accidents are independent
events, We can assume \( H_i \) and \( F_i \) are drawn from a Poisson
distribution with rate \( R_i \). We can write down the rate of all
grid cells write them down in the following vector form:

\[
|H - F|_2^2 = |H - R|_2^2 + |F - R|_2^2 + 2(H - R)(F - R). \tag{1}
\]

Since at each location \( i \), \( H_i \) and \( F_i \) are independent draws
from the same Poisson distribution, in expectation, \( F \) and
\( H \) have orthogonal deviations from the distribution mean
\( R \). Therefore, the last term in Equation 1 is negligible in
practical settings. Also \( H_i \) and \( F_i \) have similar expected
errors. Simplifying, we get:

\[
|F - R|_2^2 = |H - R|_2^2 = \frac{1}{2}|H - F|_2^2. \tag{2}
\]

Even though we don’t know \( R \), we can approximate the
error of \( F \) with respect to \( R \).

Our goal is to estimate the underlying risk of accidents
\( \hat{R} \). Since \( R \) is not given, we use \( F \) as a proxy and calculate
\( |\hat{R} - F|_2^2 \) as prediction error. There is a similar relation
to equation 1 between \( \hat{R} \) and \( F \):

\[
|\hat{R} - F|_2^2 = |\hat{R} - R|_2^2 + |F - R|_2^2 + 2(\hat{R} - R)(F - R). \tag{3}
\]

Note that in equation 3 the last term multiplies two residuals
from \( R \). If these two residuals have any significant correla-
tion, it means that our prediction shares some error with the
test set. This is not possible because our model doesn’t get
feedback from the test set. Therefore, these two residuals
should be uncorrelated in practical settings. Using Equations
2 and 3 we have:

\[
|\hat{R} - R|_2^2 = |\hat{R} - F|_2^2 = \frac{1}{2}|H - F|_2^2. \tag{4}
\]

This way we eliminate the error of \( F \) from our evaluation.

If the residuals were I.I.D., the dot product between
residuals would reach zero with a rate of \( \Theta(\frac{\sqrt{n}}{n}) \). Even
though the residuals are not I.I.D., in practice, accident
maps span a diverse area and accident rates are bounded;
therefore we can argue that the last term in Equations
1 and 3 grow slower than \( \Theta(1) \) and reach zero when \( n \) is large.

We tried formulating the same logic using maximum-
likelihood, KL-divergence and \( L_1 \)-norm. However, the
properties of orthogonality that make this analysis work are
only available in \( L_2 \)-norm.
3. Learning to predict risk maps

We use a deep model to predict accident risk at every grid cell on the map. In the design of this model, we need to overcome the challenges caused by sparsity while making sure that our model can learn useful information from all input sources. We illustrate our model architecture in Figure 3. The model takes different data modalities as input and predicts a 2D risk map \( y \in \mathbb{R}^{N \times N} \), where \( N \) is the dimension of the target map grid, and we set it to 48. Next, we discuss the details of our model.

Model Inputs: Our model takes five different data sources as input to predict a risk map for an \( N \times N \) map grid with a resolution of 5 meters. The first input is an RGB-channel satellite image. We use a higher-resolution \( (8N \times 8N) \) imagery to capture more visual information. In this case, the satellite image input is represented as an \( 8N \times 8N \times 3 \) tensor. The second input is a segmentation mask of the road map in the target region. Similar to the satellite image input, we use a high resolution for the road mask \( (8N \times 8N \times 1) \).

Our model also takes GPS trajectories as input. We represent the GPS trajectories in two formats: One is a 2D histogram \((N \times N \times 1)\) which encodes the density of the GPS trajectories (at log-scale) on each grid cell. The other format extends the 2D GPS histogram with 12 additional features, encoding the statistics (10-th, 50-th and 90-th percentiles) of the speeds, accelerations, turning angles, and the counts of left/right/no turns of all the GPS trajectories that pass through each grid cell. Combined with the original 2D GPS histogram, this input format contains 13 channels in total, and we represent it as an \( N \times N \times 13 \) tensor.

The last input source is the historical accident data. We use the rate of historical accidents in each grid cell. During training, we randomly drop out each input source with a probability of 20%. We find that this strategy is helpful when the training dataset is small. Our model supports using a subset of the above five data sources as input. In this case, we can predict risk maps even if some data sources are not available in the region of study.

Model architecture: In our model, we first pre-process input data so that different sources of data all have the same spatial dimensions. We stack the satellite input and the road segmentation input into one tensor and pass it to a 6-layer CNN encoder which down-scales the input dimension and extends the channel width from 4 (3 RGB channels + 1 map channel) to 32. Meanwhile, we use a 2-layer CNN encoder to increase the dimensions of the GPS feature input from 13 to 30. Therefore, if we stack all the input sources together, the total number of channels add up to 64.

After pre-processing, we stack all the input sources into an \( N \times N \times 64 \) tensor and pass it to a ResNet-18 encoder. We take the feature maps after each residual block set and up-sample them so that they all have the same spatial dimensions. Then, we stack them into a \( N \times N \times 960 \) feature map and pass this feature map to a 3-layer CNN decoder.

Skip Connections and Fusion: We don’t use this 3-layer CNN decoder to predict the final risk map directly; instead, we introduce a skip connection and a fusion module to produce the final risk map. We observed that the historical data is very similar to the training target in some high-risk regions. As a result, if we directly produce the risk map using the 3-layer CNN decoder, the model relies on the historical data and ignores other data sources, ending up in a low-performance local optima. To overcome this issue, we let the 3-layer CNN decoder predict two \( N \times N \) tensors: a risk map denoted as \( y_1 \) and a gate \( g \) where \( g_{i,j} \in (0, 1) \). We use the weighted average of \( y_1 \) and another risk map prediction \( y_2 \), which only uses the historical data as the final output. Formally, we have \( y = y_1 \cdot g + y_2 \cdot (1 - g) \). This allows our model to focus on learning the residual between the historical data and the target.
Table 2. Dataset details in each city. The fact that Boston has fewer accidents explains why more features don’t always help.

<table>
<thead>
<tr>
<th></th>
<th>LA</th>
<th>NYC</th>
<th>Chicago</th>
<th>Boston</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tiles</td>
<td>813</td>
<td>458</td>
<td>282</td>
<td>319</td>
</tr>
<tr>
<td>Accidents</td>
<td>351k</td>
<td>88k</td>
<td>45k</td>
<td>33k</td>
</tr>
<tr>
<td>GPS (km)</td>
<td>3.1M</td>
<td>1.8M</td>
<td>0.7M</td>
<td>2.0M</td>
</tr>
</tbody>
</table>

Target and loss function: We temporally partition accidents into two groups: historical accidents (happened before some time t) and future accidents (happened after t). A historical accident map is given as input to let the model understand the distribution of accidents. A future accident map is given as the prediction target. We use future accidents as a proxy for the true underlying risk distribution which is unknown. The future accident map is a sparse sample from the true underlying risk distribution. Therefore, it is noisy and not ideal. However, it is useful because the sampling error in the future accident map is not correlated with the sampling error in the historical accident map (because they are independent samples). Therefore, future accident map does not carry a systematic bias from historical accidents, so it is a useful proxy. Our loss function is the mean squared error between our prediction and the future accident map.

4. Evaluation

4.1. Dataset

We evaluate our model on a dataset covering an area of 7,488 km² from four metropolitan areas: Los Angeles, New York City, Chicago, and Boston. The dataset is organized as 1,872 2km×2km tiles. For each tile, we collect satellite imagery from MapBox [1] and create the road segmentation mask using OpenStreetMap [12]. Our imagery has a resolution of 0.625 meters. We also construct the road segmentation mask with this resolution.

We use a proprietary GPS dataset collected from 2015 to 2017 in the four metropolitan areas as the source of GPS trajectories. This dataset contains a total of 7.6 million km of GPS trajectories with a 1-second sampling rate.

We use the US accidents dataset [16] that contains 4.2 million records for accidents that occurred in the US from 2016 to 2020. Each record comes with coordinates, timestamps, and a few other fields of information. We split this accident dataset into two parts containing the data from the first two years and the data from the last two years. We use the first two years’ data as historical data to feed into the model as input. We use the last two years’ data as future accidents. Future accidents are used for training and evaluation. In table 2, we summarize the amount of available data in each city that helps to compare the results from the four different cities.

4.2. Training Details

In our evaluation, we split the dataset spatially into a training set (80%), a testing set (15%), and a validation set (5%). We train our models on the training set for 50 epochs, start with a learning rate of 0.0001 and decrease it by a factor of $\frac{1}{10}$ at the 20-th epoch and the 40-th epoch. The training took 6 days on one Nvidia V-100 GPU. After training, we use the validation set to find the best model and evaluate the model on the testing set. The training/evaluation code, the output accident maps, and the instruction to download the dataset are available on GitHub.

4.3. Evaluation settings

We have two major evaluation settings: with history and without history. In the “with history” setting, we supply historical accidents to the model, while in the “without history” setting, we do not supply historical accidents to the model.

Depending on the use case, historical accident data may or may not be available to the model. If historical accident data is available and the goal is to produce an accurate accident map, then accident history data should be used as input. If historical accident data is unavailable, or if this model is being used as a recommender system or to compare hypothetical designs, then historical data cannot be supplied.

We evaluate a few variants for each of the “with history” and “without history” settings. These variants include six variants of our model, kernel density estimation, and theoretical upper-bounds for low-resolution techniques. We compare with the theoretical upper-bounds as a reference to show the effect of the resolution.

In the “with history” evaluation setting, we use two years or accident data as input, while in the “without history” evaluation setting, we do not use historical accidents as input.

4.4. Evaluation metrics

In prior works, accident map prediction is formulated either as a binary classification problem or as a regression problem. Classification-based works assign a binary label (whether any accidents happened) to each cell within a time window of interest. Then they predict a score for each cell within the time window of interest. Finally, they compare their prediction scores with the binary ground truth and evaluate their performance using the precision-recall curve and average precision.

Regression-based techniques predict the number of accidents within each cell and the time window of interest. Regression-based techniques often have a low resolution; therefore, several accidents could occur within each cell. Regression-based techniques typically use RMSE between the ground-truth number of accidents in each cell and their estimation to evaluate their regression performance.
We evaluate our model with both AP and RMSE. Figure 4-a shows our precision-recall curve for the “with history” model. Figure 4-b shows our precision-recall curve for the “without history” model. Table 3 also compare average precision quantitatively.

Traffic accident estimation techniques that perform regression use RMSE to evaluate their performance. We compared our performance with RMSE in table 3. AP and RMSE have a few notable differences. First, AP puts a higher weight on high-risk locations than RMSE. Second, AP does not distinguish between one or many accidents in a cell. AP is useful for evaluating performance in high-risk areas. RMSE is useful to evaluate overall performance.

When reading these precision-recall curves, we should note that the prior probability of the prediction target has a large effect on AP statistics. A classification task on a 10m$\times$10m map has four times higher prior than a classification on a 5m$\times$5m map. Therefore, average precision numbers on different resolutions are different and should not be compared. Furthermore, since ground-truth itself is noisy, there is an upper limit on maximum AP.

4.5. Baselines and prior work

Unfortunately, the code for most of the prior work is not available. Furthermore, each prior work has studied one separate city with private data. We use the US accidents dataset [16] that is a large scale and publicly available dataset covering the entire US. In order to compare to the prior work we perform the following:

1. Since several prior works use KDE, we implemented and evaluated KDE as a baseline. The details of KDE is presented in [23]. We tuned the parameters of KDE so that it can achieve its highest average precision.
2. Since prior works use lower resolution than we do (Table 1), they cannot pinpoint accident hot-spots. This has a profound adverse effect on their performance. The effect of low resolution (100m$\times$100m vs 5m$\times$5m) is so significant that even if the prior works are allowed to optimize their output on the actual test-set, they still under-perform comparing to our model. To compare with the prior works, we use the best theoretical possible prediction (optimized on the test set with the knowledge of the future accidents) at their resolution. We refer to this as theoretical upper-bound for their accuracy. We show that our technique outperforms this theoretical upper-bound for prior works. We use theoretical upper-bound because the code for prior works is not available and they are evaluated on different cities than ours.

3. Different prior works use different sources of data as input (Table 1). We measure the effect of different sources of data on the performance of the model. This measures the effect of the extra data that we use.

4.6. Evaluation Results

We summarize our results in Table 3 and Figure 5. We show the APs and RMSEs of different approaches under two setups – with and without historical data. Next, we discuss a few insights we learned from this experiment.

Prediction with historical data: Our model uses other data sources to improve the risk map prediction when the historical data is available. As a result, our model performs better than the KDE-based approaches that only take the historical data input into account. As shown in Table 3, compared to the KDE-based approaches, our models improve the AP by 4.87 points and reduce the RMSE by 8.8%.

Prediction without historical data: When historical data is not available, our model can still use the other data
Table 3. Comparison of AP and RMSE for different methods. The first 7 rows compare the methods without using historical data as input, and the last 7 rows compare the methods that use the historical data as input. We also show the theoretical upper-bounds for the low-resolution risk maps at rows 8-10. In this comparison, all variants of our model consistently outperform other methods on both metrics.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Average Precision (%)</th>
<th>RMSE (10^{-b})</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>LA</td>
<td>NYC</td>
</tr>
<tr>
<td>GPS Density</td>
<td>16.82</td>
<td>11.87</td>
</tr>
<tr>
<td>Road (ours)</td>
<td>19.67</td>
<td>13.36</td>
</tr>
<tr>
<td>Road+GPS-Hist (ours)</td>
<td>28.59</td>
<td><strong>17.86</strong></td>
</tr>
<tr>
<td>Road+GPS-Hist+Sat. (ours)</td>
<td><strong>28.83</strong></td>
<td>16.51</td>
</tr>
<tr>
<td>Road+GPS-Feat. (ours)</td>
<td>27.28</td>
<td>15.11</td>
</tr>
</tbody>
</table>

 Besides the benefit of the GPS data source, we also observe another important fact. Among the four cities, LA is the most unsafe city (has the highest accident density), followed by New York City, Chicago and Boston. The Spar-sity of our dataset in each city follows the reverse order. If we look at the average precision (AP) of each model (with historical data) in these four cities, we find that the models that take more information as input generally perform better in LA and New York City where accidents are less sparse. However, they perform worse in Boston, where the accidents are sparser. This fact verifies that the challenge caused by sparsity does indeed exist.

Because the GPS data contains important information about traffic patterns that can be helpful for accident risk prediction, instead of using the aggregated GPS histogram or hand-crafted statistics such as the median speed, we tried to use a Deep Set [30] to extract more information from the raw GPS trajectories end-to-end. However, we found that DeepSet actually harms the accuracy in our dataset because the provision of the extra rich features from the raw GPS trajectories greatly increases model variance and over-fitting. We observe this fact even in LA, where it has the highest accident density.

Cross-city evaluation: We evaluate the generalization ability of our model in a cross-city evaluation setup where we train a model on three different cities and test it on an unseen city. As shown in Table 4, we find our model can generalize well on unseen testing cities.

Table 4. Comparison of the average precision of city-specific models vs a cross-city model. City-specific models perform slightly better. We believe this is because each city has certain unique characteristics.

<table>
<thead>
<tr>
<th>Training cfg.</th>
<th>LA</th>
<th>NYC</th>
<th>CHI</th>
<th>BOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Same cities</td>
<td>47.67%</td>
<td>28.90%</td>
<td>32.95%</td>
<td>24.63%</td>
</tr>
<tr>
<td>Cross-city</td>
<td>47.61%</td>
<td>27.79%</td>
<td>31.33%</td>
<td>24.52%</td>
</tr>
</tbody>
</table>
Figure 5. We show the risk maps produced by the KDE approach, and our approach (with and w/o historical data), along with the 2-year future accidents (used as the target in our evaluation) and the 2-year historical accidents in the four cities. We find that our risk maps can capture the underlying risk distribution that determines the probability of future accidents at all places and do so even without any historical data. In contrast, the KDE-based approach can only highlight places where there were accidents before and fail to assess the risks at other places. For example, in New York City, the accidents happened at random intersections in those two 2-year periods. Even though it looks like our risk map has low precision because there is no accident at many high-risk intersections in a 2-year period, our model captures the underlying risk distribution — accidents happen at those intersections with a similar chance.

Hyper-parameters: Since the targets (accidents) are sparse, over-fitting is a major issue. We found that optimal hyper-parameters (including model size) highly depend on the size and sparsity of the dataset. Larger models generally perform better in larger cities. Hyper-parameters must be tuned to the input size. In this work, we focused on the logic behind model design rather than tuning hyper-parameters to one dataset.

Insight: In summary, we find that the sparsity of accidents is a major challenge in the design of an accident risk prediction model. On the one hand, we need to use more data sources and deeper architectures so that we can learn a good estimation of the accident risk. On the other hand, due to the sparsity of the accidents, using more input features and deeper models can lead to overfitting.

5. Conclusion

We presented an end-to-end deep model that predicts high-resolution traffic accident risk maps. Since accident data is sparse, sample efficiency is key for a successful accident risk estimation technique. To improve sample efficiency, we use a model that establishes the similarity between locations, not just based on proximity (as in KDE) but also on similarity in appearance. We developed a model to use satellite imagery, GPS trajectories, and road maps to achieve this. We extensively evaluated and showed that our model has state-of-the-art performance. Besides the improved performance and the useful maps we generated, our evaluations provide insights into how to achieve high performance in the face of accident data sparsity.

Future work: One potential extension of this work is to combine this work with temporal risk prediction techniques to establish a spatio-temporal accident risk model. In the simplest form, there could be independent spatial and temporal components in the model. A comprehensive accident risk model could potentially input other factors, including weather patterns, driver characteristics, driving behavior, and vehicle condition.

Accident related applications: Our model is flexible in terms of what data sources are available. Once our model is trained, we can apply it to countries where detailed historical accident data is not published. Furthermore, this model can be potentially used to compare city layout designs before construction.

Other Applications: Even though this model has been developed for accident prediction, this fundamental technique can work for similar sparse location-based problems, including 911 emergency risk maps or taxi demand maps.
References


