DriveSense: Contextual Handling of Large-scale Route Map Data for the Automobile

Frederik Wiehr
Saarland University
Germany
frewie@gmail.com

Vidya Setlur
Nokia Research Center and Tableau Software
CA USA
vidya.setlur@gmail.com

Alark Joshi
University of San Francisco
San Francisco, CA USA
alark1@gmail.com

***Abstract*—Automakers are increasingly providing connectivity enhancements for vehicles to download navigational data, as well as to upload sensor information to the cloud. Generally, while more data may be better, for the driver on-the-go, information needs to be displayed in a manner that can be comprehended rather quickly. One of the major problems with visualizing route maps is that the amount of information visualized is always the same regardless of the fact that an individual may be more familiar with the region or whether an individual is driving at varying speeds. Research has shown that complex visualizations with visual clutter can cause cognitive overload that adversely affects the performance of a user. Additionally, the attention and interaction abilities of a driver are significantly compromised in a vehicular environment.**

We propose *DriveSense*, a context-sensitive visualization system that automatically varies the GPS updates and the corresponding visualization being displayed to the user based on the speed of the vehicle as well as the familiarity of the region that the user is driving in. Based on a user evaluation, we found that subjects preferred using the automatic visualizations of route maps generated by DriveSense than the visual representations shown by a standard GPS. We also computed visual clutter for our visualizations at varying speeds and found that the clutter was significantly less for the routes displayed by DriveSense for faster speeds as compared to slower speeds.

***Keywords*-route maps; automobile; context; interface;**

I. Introduction

With the rapid growth of connectivity, automobiles are becoming one of the greatest portable consumers as well as producers of data. From the need for navigation and traffic information, to the emphasis on real-time tracking of fuel efficiency, 'big data' has become an inherent part of the driving experience. With technological advancements in navigation systems, large amounts of complex map graphics can be shown in real time during the drive. Due to their high visual complexity, a higher cognitive load from the user is often required to process the information [1]. This cognitive load often varies according to the context of the driving conditions. Google Maps [2] attempts to address some of these challenges to a certain extent by providing navigation routing, displaying points of interest (POI) in an area and facilitating interactions such as zooming in and out.

The increased prevalence of interactive touch screen interfaces in cars provides additional challenges in terms of optimizing safety, usability, and affective response. While touch screens have certain usability benefits, the interfaces demand significant visual attention from the driver. Suppose that you are traveling to an unfamiliar destination in your city to visit a friend. You know that she lives close to a popular landmark (*e.g.* a mall, tourist attraction) and have visited that landmark several times. Directions from your GPS will most likely provide a shortest or fastest route, none of which will take into account the fact that you have visited the popular landmark several times. The amount of navigational detail that you would need to get to that popular landmark would be far less than the assistance you would need when you are driving in an unfamiliar region. In terms of minimizing cognitive load, existing navigation systems only show the next action that the driver needs to perform, sacrificing context along the route.

Figure 1. The DriveSense system being used in a vehicle. Since the vehicle is stationary, the speed on the smartphone is shown as 0mph.

To address these issues, we propose *DriveSense* - an automated route map visualization system that leverages large amounts of data that analyzes driving patterns to identify regions of familiarity, and automatically customizes the route map according to the perceived familiarity of that particular driver/vehicle (Figure 1). Landmarks in our system do not have to be popular, but could be destinations that the user frequently drives to, such as a supermarket, daycare,

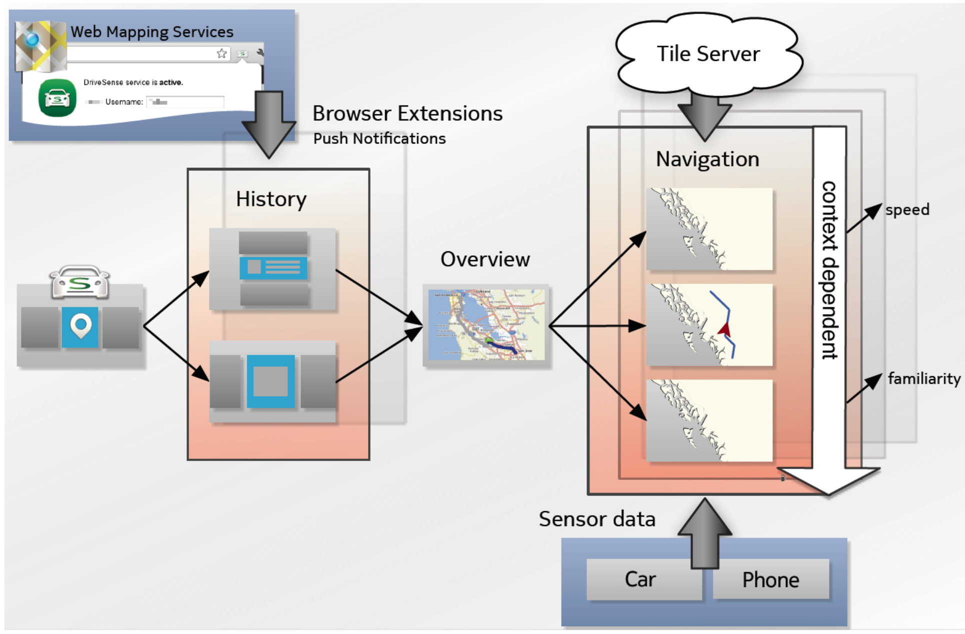


Figure 2. This figure provides an overview of the DriveSense system architecture.

work location and so on. Since the amount of detail shown to a driver can be varied automatically in our system, we can seamlessly provide context-preserving views of the route to the driver at varying speeds. Maximum detail about the route can be displayed to the driver with street names and names of nearby landmarks when the speed of the vehicle is less than 15 mph, but as the vehicle accelerates to 30 mph (the average U.S. local street speed) we display fewer details and only show nearby street names with a couple of major landmarks. As the vehicle accelerates to 60 mph (the average U.S. highway speed), we display only the major street names in a more pronounced and legible font with some minimal detail to provide context that will help the driver orient herself during navigation. This can provide cognitive cues about the overall region and may help in navigating accurately in a situation where a turn or an exit was missed.

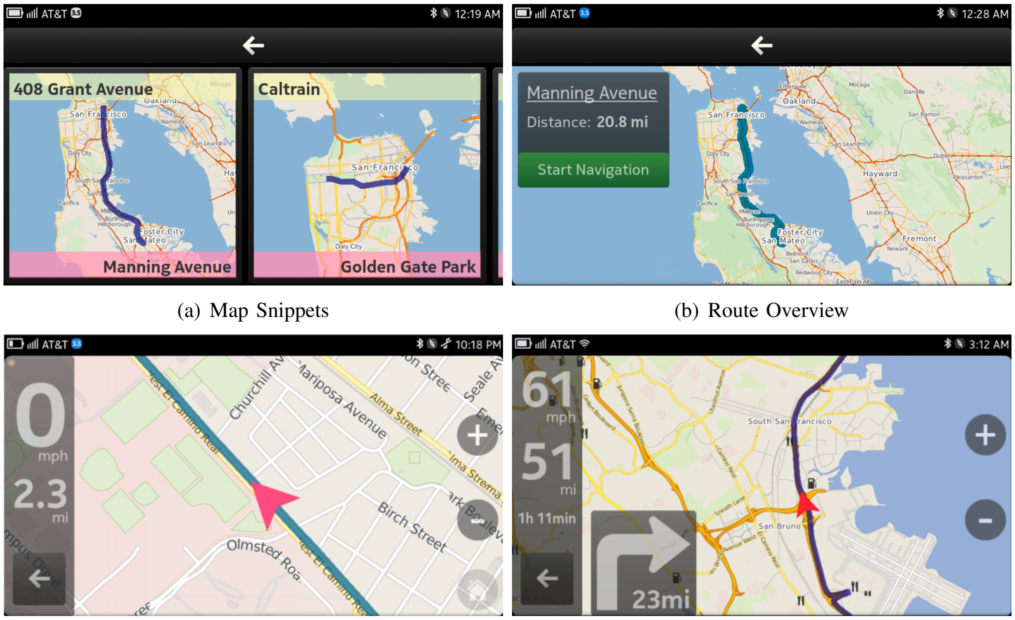
II. Related Work

To address the visualization of large networks of data, several user-driven approaches have been proposed [3], [4], [5]. Card and Nation [3] introduce degree-of-interest trees for hierarchical exploration that provides details to the user based on the user's attention (region of interest) or where one would like the user's attention to be. The ability to provide context around user-provided search words helps users explore massive graphs and networks and still find relevant information as they interact with the data [4].

Rendering route maps effectively and succinctly has become a popular research topic [6], [7]. These systems produce aesthetically pleasing results, but tend to be less useful for navigational tasks. Kray *et al.* describe a method of presenting route instructions on a mobile device depending on various situational factors such as limited resources and varying quality of positional information [8]. Reilly and Inkpen describe a map morphing as a visualization technique for relating maps with significant spatial and schematic differences [9]. Patel *et al.* [10] developed a system that suggests alternative directions based on pre-defined familiar landmarks/routes. Duckham *et al.* [11] propose a similar approach where they include popular landmarks in their routing instructions, but here too the landmarks are pre-defined. These approaches merely render an alternative route in an application and assume some prior knowledge in the system regarding the familiarity of landmarks. DriveSense aims towards learning familiarity and providing appropriate navigation directions during the process of driving to the destination. Our approach is closest to the work proposed by Dogru *et al.* [12], where they propose a user-centric approach for car-navigation. Their focus was on eliminating unimportant roads that a GPS would show, but they do not address issues of automatic generation of map visualizations based on context.

A. *Contribution*

DriveSense creates customizable maps based on contextual information obtained while driving a vehicle. The novel contribution of this paper lies in fetching appropriate routing information using sensor data obtained from the vehicle such as speed, fuel and route familiarity. Then, consequently rendering the maps by applying differential levels of detail to each object in the map based on contextual importance.



(a) Map Snippets (b) Route Overview

(c) Navigation View (d) Inset Arrow for Showing Next Step

Figure 3. DriveSense phone interface showing the different screens that a user navigates through.

III. System Overview

The system uploads web history from the desktop to the backend server through a browser extension. The extension is designed to be lightweight and integrates seamlessly with modern web browsers. The extension is triggered whenever content is loaded within the browser window. This change can be a new page load, or new content being loaded through an iframe on the web page. The extension collects the URL, title, and time of visit of the page and determines which of these web pages are related to map directions. This can be determined by examining the URL of the webpage. The extension then sends information about the map webpage such as source and destination coordinates and place information to the map server.

The map server consists of a database containing geographic data, such as roads and landmarks. Additionally, the database contains user specific data for generating personalized map layers and a routing service. Our system extracts the individual map primitives that comprise the map along with their associated importances from a database. These primitives are stylized based on map generalization principles for enhancing the most important map primitives and reducing the details of the lesser important map objects. The server then sends the map data to the mobile interface to be rendered. The overview of the system is illustrated in Figure 2.

A. Wayfinding and Map Generalization Principles

The algorithm for stylizing route maps on the server, based on driving context, takes several parameters as input: the various context parameters (such as speed, fuel and location) M_c and the importance values of the various map objects (M_i^j, where $j \in [0,n]$, n being the total number of map objects). The importance values are constrained to be $\in [0,1]$. The algorithm concerns the manipulation of individual map primitives to render maps to users as a function of semantic importance of the data represented in the map rather than traditional tile based rendering often seen in most digital map systems. The advantage of using a non-tile based rendering approach is to be able to represent specific portions of the map with greater clarity and importance than other portions. Moreover, rather than iteratively displaying the map tile-by-tile to the user, the map can immediately show portions of the map that are most relevant to the user (*e.g.* along the route from the user's current location to the desired location) and then iteratively fill in other superfluous details that are less important to the user.

In order to automate the process of visualizing route maps, we draw inspiration from a collection of perceptually based artistic approaches. Similarly, generalization is a process used by cartographers to reduce the scale and complexity of imagery while maintaining detail in important elements. These techniques facilitate differential resizing instead of a uniform scaling, involving de-emphasizing contextual objects, and increasing the detail in key objects [13].

B. Mobile Interface

The mobile user interface is built using Qt/QML and runs on a Nokia N9. The interface collects map search history from either the desktop or mobile browser. If the

(a) Underlying representation of a user's familiarity based on driving patterns.

(b) Map view showing shortest path (in red) and familiar path (in blue)

(c) DriveSense map view showing visual clutter removed along familiar path.

Figure 4. Familiarity Maps

search is done on the desktop, the URL and also the meta information such as the exact search term and if the user checked the 'start at my location' option, is sent to the phone using push notifications. The user does not have to refresh the view and in case the application is not running, it automatically pops up. Map snippets are displayed to assist the user to quickly browse and look up the map searches.The snippets are dynamic in the way that the route is recalculated according to the current location of the user. If the route is already partly traversed an arrow icon indicates where the user is along the route. Map snippets are stored locally on the phone without using a central database. The browsing interface initially shows a list in portrait mode. In this state, the snippets are accompanied with small textual descriptions. As the speed goes up, the interface changes into portrait mode it and only shows big icons with big fonts, which can be browsed horizontally (Figure 3a).

An important step before starting the navigation is to check whether the calculated route is bringing the user to the correct destination. The detail view allows a quick overview of the route that is zoomed to its bounding box (Figure 3b). At this point, all intermediate waypoints are displayed and known landmarks are shown as a first orientation aid. More detailed meta information is provided to the user in an additional overlay.

The navigation page renders custom maps generated by the server (Figure 3c). Custom per-user map layers containing personalized geo-data are rendered in realtime for every request. The sensor input from the car (through an Onboard diagnostics (OBD) Link) and the phone is collected locally and uploaded to the server whenever a good network connection is available. For a quick change of the interface, layers are preloaded and cached locally. If a condition for a layer holds, it is smoothly faded in at the right position of the stack. The interface does not deal with geographical data directly. Instead, all the necessary data to render the map is preprocessed by the server and is sent as compressed transparent images to the phone. This way, state-of-the-art map rendering and label placement algorithms can be used without having the energy and computing constraints of a mobile device.

The dashboard view indicates the current state of the car's sensor data. The current status of the connection of the car is indicated with either a red (no connection), blinking yellow (poor connection) or green icon (good connection), and common sensor data such as speed, coolant temperature, throttle position, and tire pressure.

C. *Map Stylization and Rendering*

To perform semantic zooming of map objects in the scene, it is necessary to compute the spatial detail of each object and to be able to redistribute this quantity based on importance. The spatial detail indicates the density per unit area of the vector primitives that define the map object [14].

The redistribution of spatial detail in the map is a simple budget allocation method based on the importance value of individual objects. The most important object is budgeted the largest amount of the total spatial detail available for the image, while the least important object is budgeted the least amount. The importance value of an object is constrained by definition to be $\in [0, 1]$, and the importance values of all objects are then normalized. An object cannot be made more detailed than the original or more simplified than its basic outline. As the semantic zooming increases, the number of superfluous map details such as roads and less important landmarks are removed to minimize visual clutter.

These constraints may be dictated by the physical limitations of display devices such as the size and resolution of display monitors, the minimum size and width of objects that can be displayed or the minimum spacing between objects that avoids symbol collision or overlap. A spatial detail redistribution algorithm computes a spacial detail constraint for every object to emphasize particular objects and to clarify by removing visual clutter as indicated in Algorithm 1.

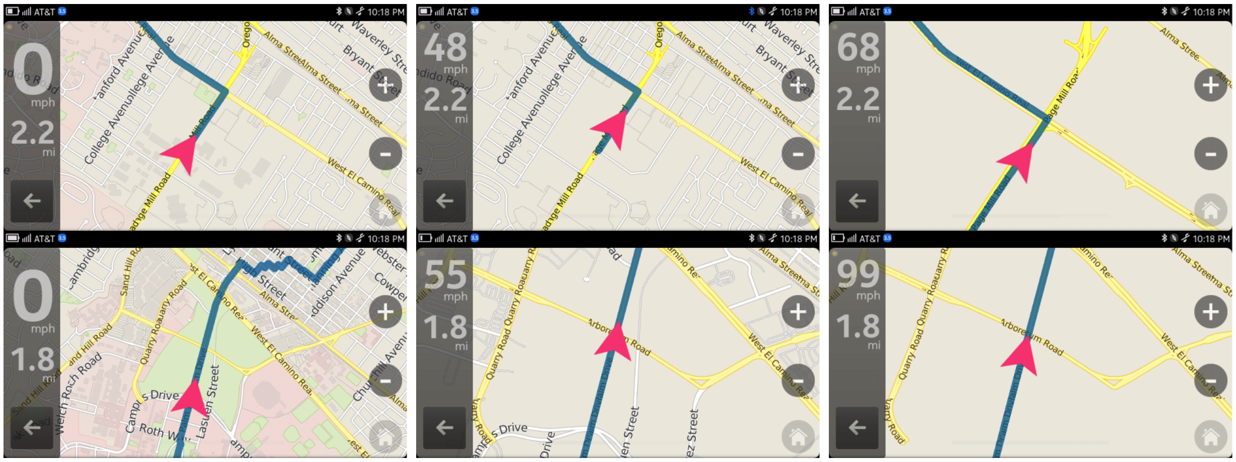


Figure 5. Comparison of route maps for varying speeds. As the speed increases (left to right), the amount of detail displayed to the user is reduced.

Algorithm 1 Spatial Detail Distribution of Map Objects

M_i is the importance of map object i, S_i is the spatial detail of map object i, S_{total} is the total spatial detail of the map, and S'_{total} is the new recomputed spatial detail after map generalization principles are applied to the map.
$M_i \leftarrow$ *Importance Value* of each map object.
$S_i \leftarrow$ Spatial detail of map object.
$Norm(S_i) = S_i / BoundingArea$
$S_{total} = \sum Norm(S_i)$
Apply exaggeration to important map objects based on car context M_c.
Compute the new total spatial detail of the map, S'_{total}
while $S'_{total} > S_{total}$ **do**
  Apply elimination, typification and outline simplification to less important map objects.
end while

1) Familiarity Maps: Matching GPS trajectories on a set of map objects such as roads is known as the map matching problem [15]. However, it does not completely apply to the collection of familiarity data, since there are less severe timing constraints. For rendering purposes and for routing, the data is aggregated as supposed to following one single object on a timeline. Also, the tolerance of false positives is much higher because a route segment or a place is only marked as familiar if it is traversed or visited frequently.

First, DriveSense collects usage data as sequential entries $(l_n1, l_n2, ..., l_m)$. The format of the log entries are $l.latitude$, $l.longitude$, $l.speed$, $l.direction$, $l.accuracy$ and $l.time$. Secondly, the system builds an index I for the log entries $l_n1, .., l_n$ for fast lookup. The algorithm now calculates the sparse matrix of the log entries being imported. Before matching single log entries against map objects, the amount of map objects is reduced by applying a bounding box filter around the total extent of the map objects. For every log entry l, it is now checked if it matches against the geometry of the map object according to the following function: $intersects(buffer(map), radius), (l.latitude, l.longitude))$, where $buffer$ creates a new geometry out of its input by adding radius as a border around it. *Radius* is determined by speed. The underlying concept is that the user can visually process less information as he travels at a faster speed along a route. If the user is slow going at only 30 mph, it is more likely that he remembers one particular landmark or a favorite place. If the condition is met, the familiarity value of a map object is increased by f which is 1.0, the highest possible.

In the last step, the familiarity values are normalized by determining the maximum value and dividing all other values by the same. At the point of rendering the map, this value is easily accessible and serves as a meaningful indicator for familiarity. Furthermore, this value is used in the cost function of the routing algorithm. The default cost function for applying the shortest path algorithm is the length. DriveSense can reduce the cost of road segment by multiplying with the familiarity measure of a single road segment. This results in routes which may be slightly longer but more familiar to the user.

2) Speed Variant Maps: Speed is used to change the level of detail in the map. The faster the user goes, the less level of detail is shown by the interface. More precisely the amount of information which is visually represented on the map is reduced as the user is going fast. DriveSense, reduces the map detail by replacing and disabling transparent map layers based on our map stylization algorithm. DriveSense modifies the zoom level of the map resulting in a constant map movement, which results in less distraction.

IV. User Evaluation

In order to evaluate the usability and effectiveness of DriveSense, we performed a user evaluation with 7 subjects.

The user evaluation consisted of driving to an unfamiliar destination using a commodity GPS and DriveSense. In order to evaluate the efficacy of using GPS as well as DriveSense to route a user to an unfamiliar destination, we conducted a survey of the subjects after having used both the GPS and DriveSense to navigate. The survey contained subjective questions pertaining to their experience. We asked them explicitly for their confidence and their perception of the ease of use of a standard GPS and DriveSense. The answers were recorded on a 1-5 Likert Scale as appropriate and longer answers were recorded when possible. Some of the questions in the survey were geared around identifying their overall perception of the neighborhood they were driving in and whether the navigation system was conveying that information to them. The survey questions were:

1) How confident were you when using a standard GPS? 1 (Not at all confident) - 5 (Very confident)
2) How confident were you when using DriveSense? 1 (Not at all confident) - 5 (Very confident)
3) Did you notice the change in detail of the display map as the car speed changed? If yes, what did you think about it?
4) If you made any errors using the standard GPS, how easy was it to navigate using the new route? 1 (Very hard) - 5 (Very easy)
5) If you made any errors using the standard DriveSense, how easy was it to navigate using the new route? 1 (Very hard) - 5 (Very easy)
6) What were the major concerns when using a standard GPS?
7) What were the major concerns when using DriveSense?
8) Were you aware of the part of town/city that you drove in when using the GPS? If so, which part of town/city was it?
9) Were you aware of the part of town/city that you drove in when using DriveSense? If so, which part of town/city was it?
10) Between a standard GPS and DriveSense, which one do you prefer better for navigation to unknown destinations?
11) Are there any other suggestions for improvement to DriveSense?

Based on the results of the survey, *all* 7 participants preferred DriveSense over a commodity GPS. The quantitative results obtained from the confidence and the ease-of-use questions were analyzed. Based on the results, the confidence of the subjects when using DriveSense was higher than when using a GPS. Similarly, they rated the ease of use to be higher when using DriveSense than when using a GPS. Figure 6 shows a plot of the two quantities for comparison. These results were analyzed using a student's t-test and were not found to be statistically significant.

Since, we wanted to find out whether the subjects noticed the change in detail of the route map and their opinion about it, we asked them an open ended-question about the same. All the subjects said that they noticed the change in detail and 6 out of the 7 subjects said that they liked the change in detail views since they were very useful and subtle. One of the subjects would have liked to have less detail even for the stationary view (for example, when he was at a traffic light). A user said that when driving at fast speeds, she would like to quickly glance at the display and DriveSense provide just the right amount of information for navigation. One of the favorite features of all the users was the automatic display of gas stations for the situation when fuel was low. They liked the fact that the maps would be personalized based on driving patterns. Since this user evaluation was conducted on a small scale, large familiarity patterns were not available and evaluated.

The major concerns and complaints users had when using a GPS was its lack of context. A subject said that if he needs to find the nearest gas station (low fuel), you would have to either stop and interact with the GPS and wait for it to find the gas station or risk driving and entering information at the same time. Similarly, a user mentioned an important concern about current GPS which is around entering destination information. One need the exact address and often takes too much effort to use it. In DriveSense, due to our integration with a browser, we cache all your destination addresses and provide "favorite" style snippets and allow for auto completion of destination address. Another concern was about how a GPS always shows the shortest/fastest path, but does not provide any contextual information for exploration of new parts of town. One of the users said that "GPS would be better if it was connected to car context like DriveSense."

In terms of suggestions for improvements of DriveSense, there were a few subjects who would have liked voice integration at various levels (provide audio cues as the map detail changes, specify address verbally and so on). Some would like to further personalize the maps and specify even less detail for stationary/slow and even medium driving speeds. In terms of personalizing the maps further, some subjects said that they would have preferred maybe a bigger display with bigger fonts to be more legible at faster speeds. This could be explored by the use of a tablet in the vehicle for navigation purposes.

V. Visual Clutter

Visual clutter has shown to cause significant degradation in performance when performing tasks [16]. Many studies have shown that a user's driving performance is adversely affected with visual clutter [17], [18]. Recently, Rosenholtz *et al.* presented a novel technique for measuring visual clutter in an image [20]. They presented two metrics to compute visual clutter: Feature congestion measure and Subband Entropy measure. The intuition behind the *Feature congestion measure* is based on practical observations. The idea is to add a feature into an image and measure the

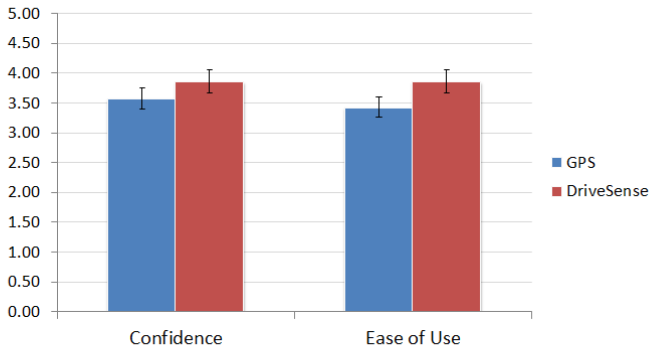


Figure 6. Graph depicting the results of the user evaluation for a standard GPS as compared to DriveSense. The average confidence was higher for subjects using DriveSense. Similarly, the average ease of use too was higher for DriveSense.

new visual saliency [21] of the image with that feature added in. If the initial image is too cluttered, then adding that feature will not result in the visual saliency to change significantly whereas if the image is neatly organized, then adding that feature will alter its visual saliency significantly. The Subband Entropy measure is based on their observation that if an image is neatly organized into groups then the image has a smaller footprint on the disk when stored as a JPEG. It performs wavelet based decomposition to compute the total clutter in the image.

We measured the Feature Congestion (FC) measure and the Subband Entropy (SE) measure for visualizations generated by DriveSense at slow speeds (0-10mph), city speeds (25-40mph) and highway speeds (60-80mph). The top row in Figure 7 shows two such input visualizations for 0mph and 60mph. The bottom row shows the output of the feature congestion step along with the FC scalar value. Based on sets of images that we tested, we found that the reduction in visual clutter was statistically significant for both the measures (FC: $p < 0.0030$, SE: $p < 0.0019$). Tables I and II show the results of analyzing the FC and SE values for the three different categories of visualization (slow, medium/city, fast/highway). Figure 8 shows a graph that shows the drop in visual clutter for both the metrics as the vehicle speed gets faster.

Source of Variation	Sum of Sq.	d.f.	Mean Squares	F
between	14.93	2	7.464	8.768
error	12.77	15	0.8512	
total	27.70	17		

Table I
FEATURE CONGESTION MEASURE: THIS TABLE DISPLAYS THE RESULTS OF AN ANOVA OF THE FEATURE CONGESTION MEASURE ON DRIVESENSE MAPS GENERATED FOR THE SAME REGION WITH DIFFERENT SPEEDS (0MPH, 30MPH AND 60MPH). THE PROBABILITY OF THIS RESULT ASSUMING THE NULL HYPOTHESIS IS $p < 0.0030$.

Source of Variation	Sum of Sq.	d.f.	Mean Squares	F
between	2.606	2	1.303	9.807
error	1.993	15	0.1328	
total	4.598	17		

Table II
SUBBAND ENTROPY MEASURE: THIS TABLE DISPLAYS THE RESULTS OF AN ANOVA OF THE SUBBAND ENTROPY MEASURE ON DRIVESENSE MAPS GENERATED FOR THE SAME REGION WITH DIFFERENT SPEEDS (0MPH, 30MPH AND 60MPH). THE PROBABILITY OF THIS RESULT ASSUMING THE NULL HYPOTHESIS IS $p < 0.0019$.

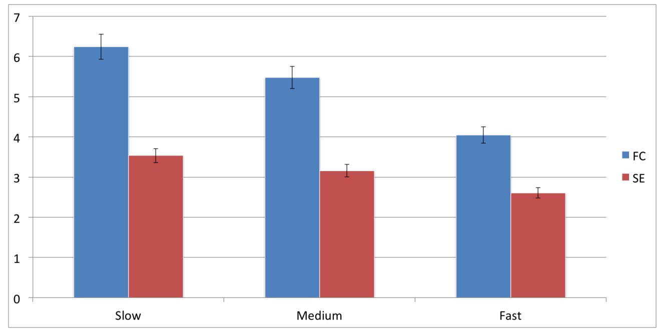


Figure 8. A comparison of the visual clutter metrics (Feature Congestion measure and the Subband Entropy measure) for visualizations displayed by DriveSense at slow, medium and fast speeds.

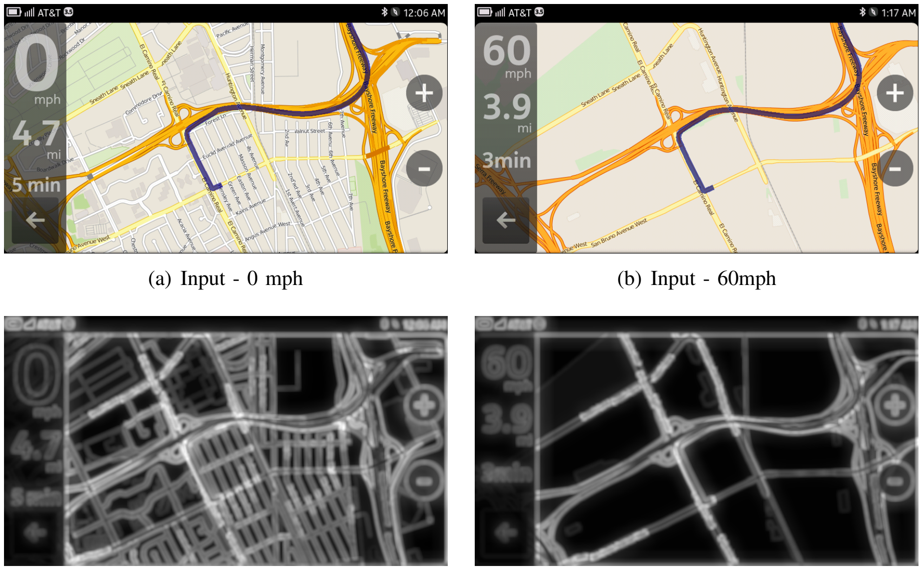
VI. CONCLUSIONS AND FUTURE WORK

The automobile has a unique role of being an active producer and consumer of big data. Despite the advancements of automotive technology, drivers are easily distracted by too much visual data. Hence, human-auto interaction is very much a work in progress. This creates unique opportunities for research in human-computer interaction for balancing data delivery and cognitive overload. As a first step in this direction, we present a system called 'DriveSense' that uses sensor information from the vehicle to vary the amount of detail being displayed to a driver.

Based on the user evaluation, we found that all the users preferred DriveSense over a commodity GPS for navigation purposes. The ability to change detail automatically in a subtle manner was most preferred. Users also liked that DriveSense could automatically sense vehicle parameters and show appropriate nearby destinations (low gas - gas station, low tire pressure - service stations) and so on. Additionally, we also measure visual clutter in our visualizations and found that the amount of detail (clutter) shown for faster speeds is statistically less than the amount of detail shown for slower speeds. A more detailed user evaluation with subjects using DriveSense for a long period of time will give us more insight into the usage patterns and the benefits of using context for navigation purposes.

REFERENCES

[1] B. Tversky, M. Agrawala, J. Heiser, P. Lee, P. Hanrahan, D. Phan, C. Stolte, and M. Daniel, "Cognitive design prin-



(a) Input - 0 mph

(b) Input - 60mph

(c) Feature congestion = 7.4626

(d) Feature congestion = 5.2772

Figure 7. The top row shows a single set of images that were measured using the feature congestion metric. The bottom row shows the resulting images and their respective Feature Congestion (FC) measures. The right column shows the map when less detail is displayed due to a higher speed (60mph). The FC measures correlate with the visual apparent reduced visual clutter.

ciples for automated generation of visualizations," *Applied spatial cognition: from research to cognitive technology*, pp. 53–75, 2006.

[2] "Google maps," http://maps.google.com.

[3] S. K. Card and D. Nation, "Degree-of-interest trees: A component of an attention-reactive user interface," in *Proceedings of the Working Conference on Advanced Visual Interfaces*. ACM, 2002, pp. 231–245.

[4] F. Van Ham and A. Perer, "search, show context, expand on demand: Supporting large graph exploration with degree-of-interest," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 15, no. 6, pp. 953–960, 2009.

[5] T. Crnovrsanin, I. Liao, Y. Wu, and K.-L. Ma, "Visual recommendations for network navigation," in *Computer Graphics Forum*, vol. 30, no. 3. Wiley Online Library, 2011, pp. 1081–1090.

[6] M. Agrawala and C. Stolte, "Rendering effective route maps: improving usability through generalization," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. ACM, 2001, pp. 241–249.

[7] A. Zipf, "User-adaptive maps for location-based services (lbs) for tourism," in *9th Int. Conf. for Information and Communication Technologies in Tourism (ENTER 2002)*, 2002, pp. 329–337.

[8] C. Kray, C. Elting, K. Laakso, and V. Coors, "Presenting route instructions on mobile devices," in *IUI '03: Proceedings of the 8th international conference on Intelligent user interfaces*. New York, NY, USA: ACM, 2003, pp. 117–124.

[9] D. Reilly, M. Rodgers, R. Argue, M. Nunes, and K. Inkpen, "Marked-up maps: combining paper maps and electronic information resources," *Personal Ubiquitous Comput.*, vol. 10, no. 4, pp. 215–226, 2006.

[10] K. Patel, M. Chen, I. Smith, and J. Landay, "Personalizing routes," in *Proceedings of the 19th annual ACM symposium on User interface software and technology*. ACM, 2006, pp. 187–190.

[11] M. Duckham, S. Winter, and M. Robinson, "Including landmarks in routing instructions," *Journal of Location Based Services*, vol. 4, no. 1, pp. 28–52, 2010.

[12] A. Dogru, C. Duchêne, S. Mustière, and N. Ulugtekin, "User centric mapping for car navigation systems," in *11th ICA Workshop on Generalisation and Multiple Representation*, 2008, pp. 20–21.

[13] C. Board, "Map reading tasks appropriate in experimental studies in cartographic communication," *Canadian Cartographer*, vol. 15, no. 1, pp. 1–12, 1978.

[14] M. Amadasun and R. King, "Textural features corresponding to textual properties," in *IEEE Trans. Sys. Man, Cybern.*, vol. 19, no. 5, 1989, pp. 1264–1274.

[15] J. Krumm, J. Letchner, and E. Horvitz, "Map matching with travel time constraints," in *SAE World Congress*, 2007.

[16] M. Yeh, J. Merlo, C. Wickens, and D. Brandenburg, "Head up versus head down: The costs of imprecision, unreliability, and visual clutter on cue effectiveness for display signaling," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 45, no. 3, pp. 390–407, 2003.

[17] W. Horrey and C. Wickens, “Driving and side task performance: The effects of display clutter, separation, and modality,” *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 46, no. 4, pp. 611–624, 2004.

[18] S. Baldassi, N. Megna, and D. Burr, “Visual clutter causes high-magnitude errors,” *PLoS Biology*, vol. 4, no. 3, p. e56, 2006.

[19] J. Henderson, M. Chanceaux, and T. Smith, “The influence of clutter on real-world scene search: Evidence from search efficiency and eye movements,” *Journal of vision*, vol. 9, no. 1, 2009.

[20] R. Rosenholtz, Y. Li, and L. Nakano, “Measuring visual clutter,” *Journal of Vision*, vol. 7, no. 2, 2007.

[21] R. Rosenholtz, “A simple saliency model predicts a number of motion popout phenomena,” *Vision research*, vol. 39, no. 19, pp. 3157–3163, 1999.