

# SideEye: Mobile Assistant for Blind Spot Monitoring

Sanjeev Singh, Rufeng Meng, Srihari Nelakuditi, Yan Tong, Song Wang  
University of South Carolina, Columbia  
{singhsk, meng, srihari, tong, songwang}@cse.sc.edu

**Abstract**—Distracted driving is an ever growing concern and driver safety systems are increasingly getting adopted. However, they still remain as features in the luxury vehicles. The high penetration of smartphones has made it possible to bring some of those safety features within everyone’s reach. While some of the recent works have proposed tracking the road conditions as well as monitoring the driver with a smartphone, they have not covered the blind spot. In this paper, we present SideEye, a smartphone based system to monitor the blind spot on the driver side and alert the driver about the presence of a vehicle. We explore two approaches based on intensity variation and contour matching to detect a vehicle in the blind spot. Our evaluation shows that, when a vehicle is in the blind spot, our system can detect and alert the driver with an accuracy of 87% in real-time.

## I. INTRODUCTION

According to the traffic safety facts report of National Highway Traffic Safety Administration, more than five million police-reported motor vehicle crashes occurred in the United States in 2011 [1]. One of the causes of these accidents is that drivers change lane without checking the blind spot. The common solution for this problem is to use blind spot mirrors. But often mirrors are not set properly. Furthermore, they work in a passive manner. If the driver does not check it frequently, he could still miss what’s happening in the blind spot. Therefore, apart from advocating defensive driving, developing and deploying technologies to actively monitor driver’s blind spot and alert driver to be aware of the vehicle in his blind spot is essential to avoid collisions.

Blind Spot detection systems are offered by many vehicle manufacturers, but they are not available in all models. These systems are typically a luxury feature and usually a part of an expensive safety package. It will take decades before such features become commonplace. So how do we bridge this safety gap? The answer could be smartphone-based systems. Nowadays smartphone has become ubiquitous and all-purpose. More than 60% mobile phone subscribers in the US are using smartphones [2]. People are already using smartphones in their vehicles for navigation. Smartphones contain most of the necessary and similar functional components as commercial systems, i.e. screen, camera (usually one front, one rear), speaker, etc. Instead of being a cause of driver distraction, recently smartphones have become the focus of improving driver safety. Some of the functions of the commercial safety systems are being implemented on smartphones to assist drivers.

Blind spot is usually the area the driver is least informed about and hence any assistance in monitoring blind spot can greatly enhance driver’s safety. Therefore, we develop a smartphone-based application, referred to as *SideEye*, for detecting the presence of vehicles in the blind spot and alerting the driver. SideEye provides an affordable alternative for

bringing blind spot monitoring function, which approximates the similar safety feature of luxury vehicles to economy vehicles. Although it might not be a complete substitute for the sophisticated safety features in the luxury vehicles, it goes a long way in making driving safer.

To be a practical driver safety system, SideEye should identify the situation very accurately and alert the driver in real-time. In SideEye, we use computer vision related technologies to analyze the scene in the blind spot area. There are several challenges in realizing SideEye. First, compared to dedicated cameras used in other safety systems, the resolution of the smartphone camera is not so high in recording video, especially the front camera that is watching the blind spot. And the video frames captured by the smartphone camera are heavily affected by the light condition and the dynamic environmental factors. The low-quality frames will make it difficult to correctly identify what is really happening in the blind spot area. Second, computer vision technology is usually computationally heavy. Obviously, compared to a desktop or a laptop, the power of CPU in a smartphone is lower and memory is smaller, and few smartphones are equipped with GPU. So when processing the frames, it will take more time on the smartphone, which might delay the alert, making it ineffective. In this paper, we address these challenges. We explore two computationally efficient approaches based on intensity variation and contour matching to detect vehicles in the blind spot. Our evaluation shows that these approaches hold promise, encouraging us to further refine them and develop a robust system for monitoring blind spot.

The rest of the paper is organized as follows. In Section II, we present the related work in the driver safety area. Section III gives an overview of the system. We describe our design and implementation in Section IV and report its performance evaluation results in Section V. We list the limitations of current version of SideEye and discuss how we can improve it in Section VI. Section VII concludes the paper.

## II. RELATED WORK

There has been extensive work done on making driving safer and commercial systems already exist. BMW has introduced Active Blind Spot Detection system. Volvo uses a sensor-based system that provides the driver with a warning if a vehicle enters his blind spot while he is changing lanes. Mercedes and many other OEMs also have their own blind spot monitoring systems. Although there is a need for these safety features, they are limited to luxury vehicles only. A major section of vehicles are still without this safety feature.

There are also many aftermarket products for blind spot detection which offer a choice over the features offered by luxury vehicles. Mobileye [3] developed a blind spot/lane

change assist system which mounts a camera on the side mirror to monitor the vehicles moving behind on the next lane. Iteris [4] offers blind spot warning system. Both Mobileye and Iteris use specialized hardware. However, the market penetration for such devices remains low, especially in low-end vehicles. This is where the smartphone offers a viable solution for taking the safety technology to most of the drivers. The import of prior driver safety related work to the smartphone platform will make it useful for many.

In academia, lot of research has been done to improve driver safety. VioLET [5] uses vehicle-state information and cameras to track lanes, monitor head pose and lane changing intent. Several similar research proposals utilize specialized hardware. Recently, there has been active work on using smartphones to assist drivers. SignalGuru [6] advises the driver to maintain a certain speed while approaching a signal for fuel efficiency. The works in [7] [8] proposed smartphone-based systems that monitor both the driver and the road ahead by its dual cameras. The system in [8] detects driver distraction, vehicle proximity and weaving. In SideEye, we focus on blind spot detection and improve the safety in lane changing.

### III. SIDE EYE OVERVIEW

Although blind spots are present on both sides of the vehicle, we propose to monitor only the driver side. A collision on the driver side can be more damaging to the driver and in most instances there are no other passengers in the vehicle. Driver-side blind spot is also difficult to check by turning head, as there is little space between the driver and the blind spot on his side. Even by turning head, driver's view is always blocked by the pillar between the front door and back door, especially the driver's head is so close to that pillar on his side. In this section, we describe how SideEye operates to detect a vehicle in driver-side blind spot and alert driver.

#### A. Area to Monitor and Alert

Fig. 1 shows an example of how we mount the smartphone in the vehicle. We can easily find a place to mount the smartphone on the windshield or on the dashboard to allow the front camera of the smartphone to monitor the blind spot area. The red polygon shown in the right of Fig. 1 is the region of interest (ROI) on which our SideEye system will monitor, which covers the blind spot of the driver. The ROI is the area on the next lane on the driver side. When there is a vehicle in ROI, it is easy to cause collision if the ego-vehicle attempts to change lane without seeing that vehicle.



Fig. 1. Mounted position of the smartphone and the region of interest

The smartphone's screen is facing the driver. Whenever SideEye detects a vehicle in the ROI, it alerts the driver as soon as possible to improve safety. The system can work as a

daemon thread simultaneously with other apps in smartphone, e.g. navigation app. Only when the system detects something critical, i.e. a vehicle is in the ROI, it alerts the driver. The system could alert the driver in different ways, for example, show a warning icon on the screen or give an audio warning.

#### B. Schemes

We use camera to monitor the ROI and process the information in the video frame. Naturally, we employ computer vision based techniques to do this job. We explore two approaches to detect whether there is a vehicle in the ROI. One checks for the change in the intensity of image within ROI; the other looks for the vehicle in ROI by the knowledge of vehicle's shape. Below, we present the intensity and contour matching based schemes for detecting vehicles.

1) *Intensity based scheme:* When there is a vehicle in the ROI (as shown in the top-right of Fig. 2), the distribution of the pixel intensity of the ROI could be significantly different from the scenario when the road is empty (Fig. 2 (top left)).

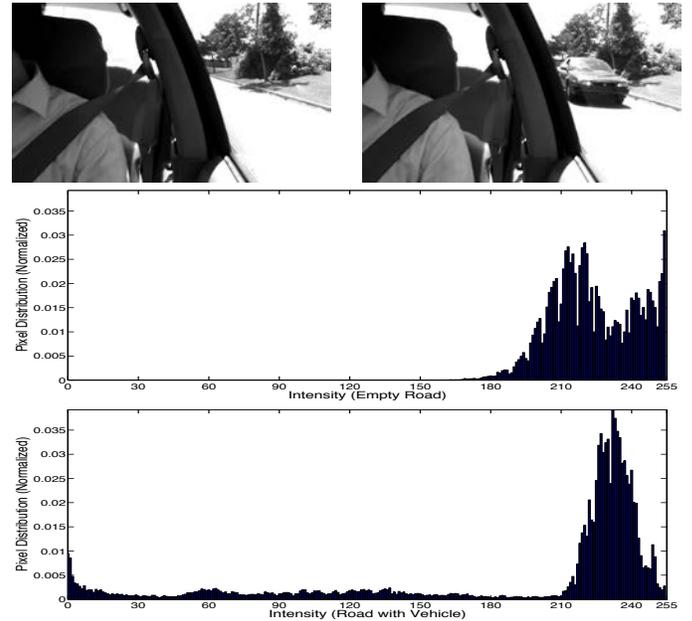


Fig. 2. Empty road (top left) and road with vehicle (top right); the corresponding pixel intensity distribution for empty road (middle) and for road with vehicle (bottom).

We utilize the difference between the pixel intensity distribution to identify whether there is a vehicle or not. In the following, we propose two sub schemes based on the intensity variation.

a) *Intensity Variation:* One way to distinguish between an empty road and a vehicle, is to observe the intensity variation of the region. It is based on the idea that the intensity variation of an empty road will be small compared to that of a road with a vehicle. We calculate the intensity variation and use a threshold to separate empty road from non-empty road. Usually an empty road will have most of its pixels concentrated around one peak (shown as in Fig. 2 (middle)). When the texture of the road changes, the peak may shift, but most of the pixels are still concentrated around that. When a vehicle enters the region, there will be a spread of pixels and other

local peaks will occur (as shown in Fig. 2 (bottom)). We use this characteristic as a hint to tell whether a vehicle is present in the region.

*b) Intensity Variation with Warping:* The problem with the simple intensity variation based approach is that it calculates the intensity as seen from the camera view. A vehicle which has just entered the region will only occupy few pixels, and therefore their contribution to the overall pixel intensity distribution is small.

It is important to detect a vehicle as early as possible. For that, we warped this region so that it gives an even distribution of pixels. This way, a vehicle which has just entered the detecting region will have a proportional contribution to the overall intensity variation.

Fig. 3 shows how the warping works. The left image shows the ROI with a vehicle in it. A vehicle just enters this area and only occupies a small portion of the ROI. The right image shows what the scene will be after warping. When warped, the vehicle will occupy more in this ROI and hence it could contribute more to the overall pixel intensity distribution.

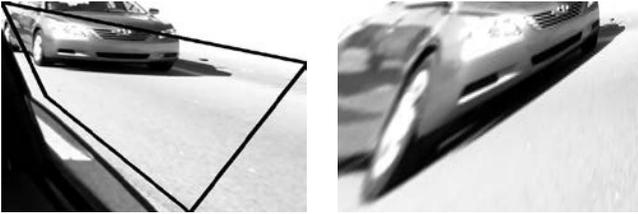


Fig. 3. ROI with a vehicle (left) and the scene after warping (right).

*2) Contour Matching based Scheme:* Vehicles have similar contours. If we can find a contour in ROI which matches the contour of vehicle, we may conclude that a vehicle is in the region. We adopt Chamfer Matching to do this work. Chamfer Matching computes the distance or dissimilarity between two images. It basically matches a template with the contours of a query image and finds the best matching contour in the image. It first calculates the edges of the template and the query image, and then computes the distance between pixels on the edges. We use this scheme for early detection of vehicles. Most of the vehicles' front contours are similar, with the wheels, hood and the roof. We draw a general template of vehicles and use Chamfer Matching to detect a similarly looking object, i.e. the vehicle. We then put a threshold on the cost/distance of matching, with the smaller cost implying a match with a vehicle (as shown in Fig. 4). Moreover, even if there is no perfect match of the template, it is still possible to detect a vehicle. Since a vehicle has many edges, its distance from the template will still be a low value compared with an empty road which does not have many edges.

#### IV. DESIGN AND IMPLEMENTATION

We need to decide the scene of the ROI, i.e. is it an empty road or is there a vehicle present in it. From Fig. 2 we know the distribution of the pixel intensity is different for empty road and road with vehicle, hence, the variation is different. So we need to find out the optimal threshold for the variation which can confidently distinguish between empty road and road with vehicle. When the road is empty, it should not report



Fig. 4. Chamfer Matching of the template to a vehicle.

as a vehicle; when there is a vehicle, it should correctly tell us a vehicle is there. We use macro-average F-measure (as shown in the following formula) to select thresholds. The one which maximizes the macro-average F-measure is the optimal threshold. (In the formulas, e = empty road; v = road with vehicle; TP = True Positive, FP = False Positive, FN = False Negative, F = F-measure.)

$$Precision_e = TP_e / (TP_e + FP_e)$$

$$Recall_e = TP_e / (TP_e + FN_e)$$

$$Precision_v = TP_v / (TP_v + FP_v)$$

$$Recall_v = TP_v / (TP_v + FN_v)$$

$$F_e = 2 * Precision_e * Recall_e / (Precision_e + Recall_e)$$

$$F_v = 2 * Precision_v * Recall_v / (Precision_v + Recall_v)$$

$$F_{MacroAverage} = (F_e + F_v) / 2$$

We build a model which tracks the macro average F-measure by choosing different thresholds. We trained this model on 18000 sample images, which includes 11600 images with vehicles and 6400 empty road images. The Fig. 5 (left) shows at which point we get the optimal threshold for the Intensity Variation scheme. We find 35 to be optimal threshold.

Similarly, for the Intensity Variation with Warping scheme, we choose 46 as the threshold of variance (as shown in Fig. 5 (middle)) to distinguish the empty road and road with vehicle.

For the Chamfer Matching scheme, first, we construct a contour template based on the general shape of vehicles, and then try matching on the training image set. Every matching has a cost. If the cost is low, we consider the subject is matched, i.e. it is a vehicle; otherwise, it should be an empty road. So here the problem is to choose an optimal cost which allows us to confidently tell whether the ROI contains a vehicle or not. We also use the macro average F-measure to decide the best threshold for the matching cost. Fig. 5 (right) shows that we can get the most confident conclusion about the ROI at the threshold 0.2.

We use the thresholds obtained above to implement our SideEye system for the three schemes in distinguishing vehicle from empty road. The flow of intensity variation based schemes is shown in Fig. 6 (left) and the flow of Chamfer Matching based scheme is shown in Fig. 6 (right).

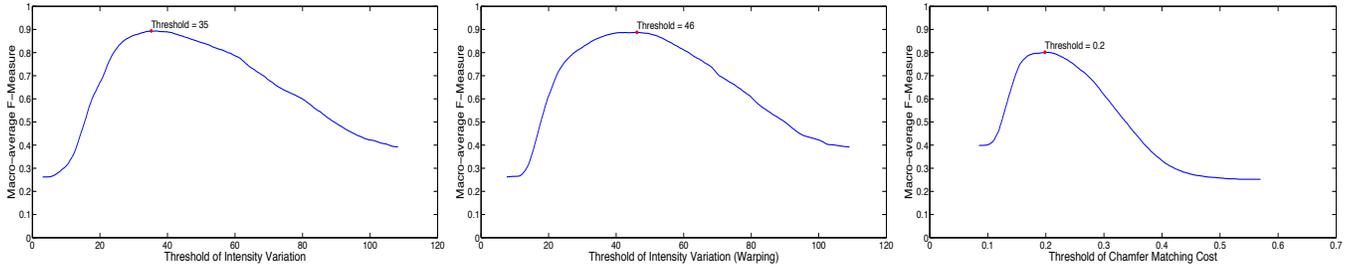


Fig. 5. Threshold selection for Intensity Variation (left), Intensity Variation with Warping (middle) and Chamfer Matching scheme (right).

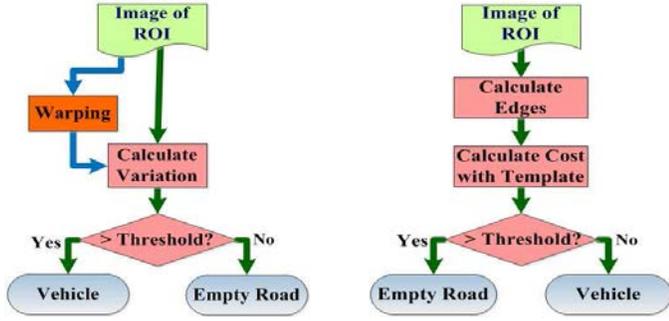


Fig. 6. (Left) Flow of intensity variation based scheme (with/without Warping). (Right) Flow of Chamfer Matching based scheme.

## V. EVALUATION

### A. Dataset and Environment

To evaluate SideEye, we tested it on 6350 video frames taken on US I-20 and US I-26. Our final target is a safety app, but for the sake of efficiency during evaluation, we tested our system in a laptop and ran on the video frames offline. The platform we use in testing will not affect the accuracy of our system, only the efficiency of our system will change from platform to platform. We measured the computational efficiency of each scheme on a real smartphone.

### B. Performance of Schemes

We measured the accuracy, precision and recall for each scheme. The results are compared in Fig. 7.

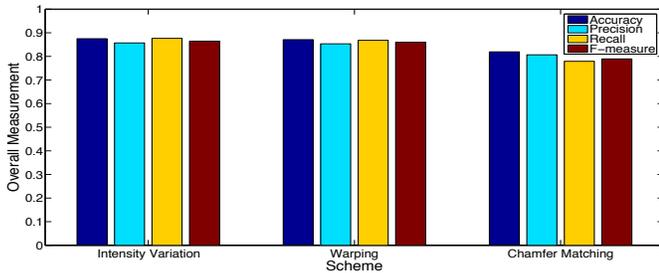


Fig. 7. Overall performance of the schemes.

Both Intensity Variation and Intensity Variation with Warping can correctly detect the scene in the ROI with an accuracy at about 87%. Chamfer Matching’s accuracy goes to 82%. As to precision and recall, the performance of Intensity Variation and Intensity Variation with Warping are also very close, and they both work better than Chamfer Matching scheme.

As a safety system, it should alert the driver when a vehicle is in the ROI. But when the vehicle is at different positions in ROI, the danger to the ego-vehicle is different. The danger increases as the vehicle moves closer. Here we break down the ROI into small segments to check how the performance will be when the vehicle is at different positions. Fig. 8 shows how correctly each scheme can identify the situation when the vehicle reaches at different positions in the ROI.

We can see that for Intensity Variation and Intensity Variation with Warping scheme, if there is a vehicle in the ROI and has moved across 20% of this area, both these schemes can detect it very accurately. And when the road is empty with no shadow, it can also correctly find out that it is empty road.

But when a vehicle just entered the ROI (i.e.  $< 20\%$ ), the two intensity variation based schemes cannot confidently tell whether it is a vehicle or not. At this moment, although the overall intensity of this area changes, the vehicle only occupies a very small portion of this area. Therefore, the overall intensity variation is not affected much, which makes the two schemes fail to detect the vehicle. But note that, when a vehicle only occupies  $< 20\%$  of the ROI, that vehicle is still at some distance from the ego-vehicle. As a result, even if the system cannot confidently alert at this moment, the driver will not be in immediate danger. Furthermore, when the vehicle gets even slightly closer in ROI, say beyond 20%, accuracy improves quickly, thus ensuring the safety of drivers.

When a vehicle is moving in the “rest” area, i.e., parallel to the ego-vehicle, it occupies less and less portion of the ROI and its influence on the overall intensity variation decreases. Consequently, the overall detection rate here is not as high as other categories where the vehicle occupies a major part of the ROI. Still, the detection accuracy is between 80% and 95%. Also, when a vehicle is moving parallel to the ego-vehicle, the driver of the ego-vehicle can see that vehicle and SideEye performance is not as critical in this case.

An empty road with shadows poses a problem for both these schemes. Fig. 8 shows that less than 30% of empty roads with shadows are correctly identified. When an empty road has shadow from buildings, trees or some other objects, the intensity of the ROI could be totally random. In some cases, the shadow makes the overall intensity variation of the ROI look like a vehicle is present. Therefore, both these schemes cannot confidently distinguish shadow from a vehicle.

Compared to Intensity Variation, Intensity Variation with Warping works slightly better when a vehicle only occupies  $\leq 30\%$  of the ROI. The reason for the difference in the performance here is that the warping increases the portion of the corner area in which the vehicle just entered. On the other

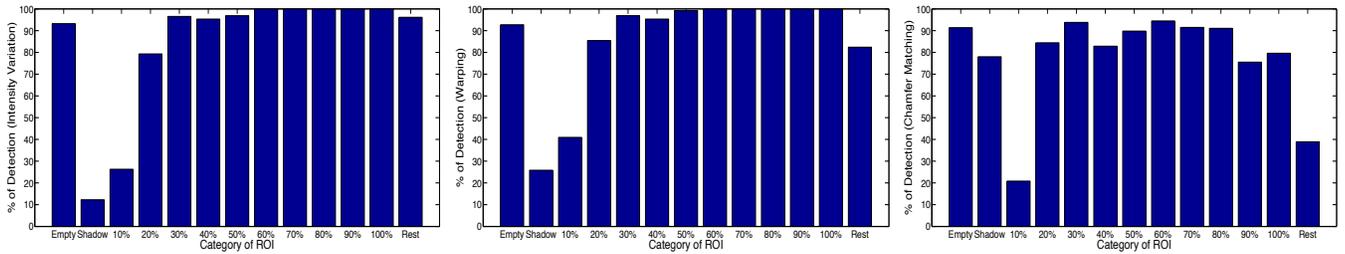


Fig. 8. Performance break-down of each scheme for different positions in ROI: (left) Intensity Variation; (middle) Intensity Variation with Warping; (c) Chamfer Matching. “Empty” means there is no vehicle in ROI; “Shadow” means there is no vehicle but shadow from buildings, trees or other things falls in ROI; 10% means the head part of the vehicle occupies 10% of the ROI at the far corner; 100% means the head part of the vehicle reaches the end-edge of ROI; “Rest” means the head part of the vehicle has passed the end-edge and only the tail part can be seen.

hand, for the “rest” area, warping increases the portion of the far corner of the ROI which makes the portion of the end-edge area decrease. As a result, in that case, Intensity Variation performs better than Warping scheme.

Now let us consider the performance of Chamfer Matching. When there is a vehicle in the ROI which occupies 20% or more of the ROI, it can detect the vehicles with about 80% accuracy. Though its performance is slightly worse than the other two schemes for the same scenario, it has a valuable feature. Different from the intensity variation based schemes, if the road is empty, regardless it has shadow or not, the Chamfer Matching scheme can recognize it much more accurately.

### C. Efficiency

As a traffic safety system, the efficiency is very important. In case of an unsafe scenario, the system must alert the driver in real-time. Here we measured how fast each scheme could be in detecting. We ran our system on a Galaxy Note and show the mean computation time on each video frame.

TABLE I. COMPUTATION TIME OF EACH SCHEME.

Scheme	Mean Computation Time Per Frame (ms)
Intensity Variation	38
Intensity Variation with Warping	53
Chamfer Matching	128

From Table.I, we can see that all the schemes can process in less than 130 ms. So our system can alert drivers in real-time. When new smartphones come out, faster multi-core CPU, larger memory or even GPU will be equipped. We can expect that SideEye can run even faster in the new smartphones, making it practical for improving driving safety.

## VI. LIMITATIONS AND FUTURE WORK

Current implementation of SideEye utilizes intensity variation based schemes. These schemes can not correctly distinguish the shadowed empty road from road with vehicle. But by looking at the performance of Chamfer Matching, we can see that the weakness of the intensity variation based scheme can be compensated by the Chamfer Matching scheme. So we plan to combine these schemes together to build a hybrid scheme, which will gain the merits of both worlds and improve the overall performance of this system.

Another aspect we plan to improve is for the ROI construction. At this moment, for the sake of easy implementation and evaluation, we manually select four points in that area to get

the ROI. In the future, we will setup the ROI automatically by detecting the lane marks on the next lane as well as the edges of window frame of the ego-vehicle on the driver side.

We will build a fully functional smartphone app to make it available for everyone. We hope that it can help the drivers to gain safer driving.

## VII. CONCLUSION

In this paper we present SideEye, a smartphone-based application which monitors the driver-side blind spot and alerts the driver. We evaluated our system on data collected from a typical highway driving, and also measured the efficiency on a real smartphone. The result explains that it is possible to effectively detect and alert the driver when there is vehicle in the blind spot area. Our system can achieve precision and recall both at about 85% in identifying the scene in the blind spot area. Although SideEye needs further refinement to make it robust, it shows promise in improving the driver safety. If done well, SideEye can fill yet another safety feature lacking in non-luxury vehicles by bringing it to the ubiquitous smartphones.

## REFERENCES

- [1] “Traffic safety facts,” <http://www-nrd.nhtsa.dot.gov/Pubs/811753.pdf>.
- [2] “Smartphone market share,” [http://www.comscore.com/Insights/Press\\_Releases/2013/9/comScore\\_Reports\\_July\\_2013\\_U.S.\\_Smartphone\\_Subscriber\\_Market\\_Share](http://www.comscore.com/Insights/Press_Releases/2013/9/comScore_Reports_July_2013_U.S._Smartphone_Subscriber_Market_Share).
- [3] “Mobileye applications,” <http://www.mobileye.com/en/manufacturer-products/applications>.
- [4] “Autovue lane departure warning,” <http://www.iteris.com/upload/datasheets/LDW/Final/web.pdf>.
- [5] Joel C McCall and Mohan M Trivedi, “Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation,” *Intelligent Transportation Systems, IEEE Transactions on*, vol. 7, no. 1, pp. 20–37, 2006.
- [6] E. Koukoumidis, L.S. Peh, and M.R. Martonosi, “Signalguru: Leveraging mobile phones for collaborative traffic signal schedule advisory,” in *Proceedings of the 9th international conference on Mobile systems, applications, and services*. ACM, 2011, pp. 127–140.
- [7] Sanjeev Singh, Srihari Nelakuditi, Romit Roy Choudhury, and Yang Tong, “Your smartphone can watch the road and you: mobile assistant for inattentive drivers,” in *Proceedings of the thirteenth ACM international symposium on Mobile Ad Hoc Networking and Computing*. ACM, 2012, pp. 261–262.
- [8] Chuang-Wen You, Nicholas D Lane, Fanglin Chen, Rui Wang, Zhenyu Tong, Thomas J Bao, Martha Montes-de Oca, Yuting Cheng, Mu Lin, Lorenzo Torresani, et al., “Carsafe app: Alerting drowsy and distracted drivers using dual cameras on smartphones,” *ACM Mobisys*, 2013.