

# Wide-scoped Around View Detection

Yu-Chung Kuo and Din-Chang Tseng\*

Institute of Computer Science and Information Engineering, National Central University, Taiwan

\*Corresponding author

**Abstract**—In recent years, around-view monitoring systems have become a public driving assistant for reducing collision hazards by eliminating invisible areas. Many of such systems provide short range views surrounding the vehicle, limiting its application to parking and reversing. We had developed a wide-scoped around view monitor system; however, only monitor is not enough for complete protection; thus, in this paper, we propose a detection system to highlight the wide-scoped around-view monitor system by detecting possible obstacles around the driving environment. We use four cameras mounted on four sides of a vehicle to capture the surrounding images; these images are then processed and projected on a dual-camber model centered by the vehicle. The projected imagery gives drivers the freedom to change view-point to suit different driving needs. By estimating the ego-motion of the vehicle using the input image sequence of the cameras, the proposed system is able to detect objects in the images by finding movements of features that do not correspond to ground motion relative to vehicle motion. Detected obstacles are highlighted in the wide-scoped around-view imagery to warn the driver of potential hazards. We tested the proposed system against asphalt, concrete, and tiled road surfaces with obstacles in the scene. The results show while concrete and tiled surface features can be effectively removed, feature-poor asphalt surface is prone to misdetection for errors introduced during calibration and ego-motion estimation.

**Keywords**—advanced driver assistance system; around view monitor; obstacle detection; homographic transformation; ego-motion estimation

## I. INTRODUCTION

Following the advancement of science and technology, improving drive safety has become of higher concern. Among the causes of traffic accidents, one main cause is due to limited field of view of the driver. The driver is unable to acknowledge the entire area surrounding the vehicle, pay attention of different directions at the same time.

In order to monitor the vehicle's surrounding; cameras are mounted around the vehicle to capture image sequences and provide the driver with a visual monitor. Many institutes have developed such systems for parking assistance; such as Nissan's "Around View Monitor" [1], Honda's "multi-view camera system" [2], etc [3-5]. Most of such systems provide short range views surrounding the vehicle, limiting its application to parking and reversing. We had developed a wide-scoped around view monitor system [6]; however, only monitor is not enough for complete protection. An active obstacle detection and warning system is helpful to enhance the protection.

Hoiem *et al.* [7] trained the machine with images of color, texture, position, shape, and geometric constraints, of up to 78 different feature types, then detected road surface, above-ground objects and sky in real-time. AdaBoost was then used to train for the detection of pedestrian and vehicles, and a probability model was proposed taking view angle, object, and environment into consideration for pedestrian and vehicle detection. Saxena *et al.* [8] also used images of color, texture, position, shape, and geometric constraints, with 646 different features in training, then computed the depth map and detect obstacles.

Bertozzi *et al.* [5] set up two cameras on a truck and by using grid points on the ground to inverse project the images to top-view images. Then polar histogram is used to analyze the difference of two images to check for object perpendicular to the ground. Enkelmann [9] detected obstacles by comparing calculated optical flow fields with a model vector field which explicitly expresses the image shift expectation from the camera motion if the environment is free of obstacles. Gandhi and Trivedi [10] proposed a system utilizing an omnidirectional camera mounted at top of the vehicle. By estimating the ego-motion using known planar motion model, the system filters out moving obstacles from the calculated optical flow and estimation the relative movement information.

In this paper, we propose a detection system to highlight the wide-scoped around-view monitor system by detecting possible obstacles around the driving environment. The proposed system employs four wide-view cameras mounted at front, rear, and both sides of the vehicle to capture images of vehicle's surrounding. After a sequential transformation and combination, the system provides a bowl-shaped wide-scale view of vehicle's surroundings, giving driver a full view around the vehicle. On the other hand, the input images are also used for detection of above-ground objects that may be hazardous to the vehicle. Detected object is labeled on the surrounding top-view to warn the driver. Detection is especially helpful for that general person may miss out obvious obstacles even when paying attention due to inattention blindness.

The complete system is comprised of three parts: offline calibration, wide-scoped view generation, and wide-scoped obstacle detection as shown in **Fig.1**. Offline calibration process includes the calibration of the camera parameters, image and distortion parameters. The combined calibration includes top-view transformation, image registration, and projection to bowl shaped model. With the offline calibration and transformation process, we create a mapping that maps input images to the wide-scoped view on the bowl shaped model. Because the cameras' positions are fixed on the vehicle,

the geometric relationship between cameras and the vehicle does not change, the mapping created at offline calibration process can be used repeatedly during the online phase.

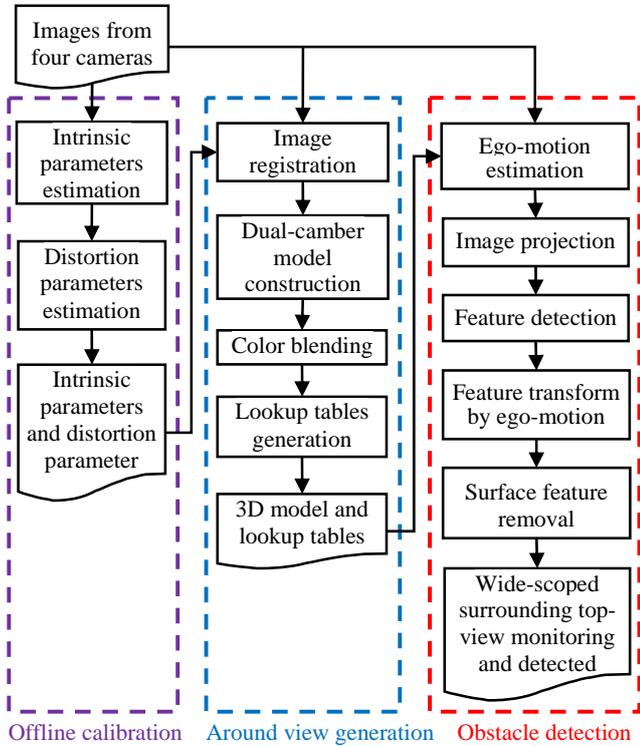


FIGURE I. WIDE-SCOPED AROUND VIEW MONITOR AND DETECTION SYSTEM.

Wide-scope surrounding top-view monitoring (WSTM) projects the input images onto the bowl shaped model according to the mapping produced during offline process. The projection includes interpolation and color blending.

Detection of above-ground objects is done during online process in real-time. The process includes ego-motion estimation, object detection, and removal of surface features as described in Fig.2. Ego-motion of the vehicle is estimated by solving the translation and rotation from matching features between input images from two consecutive time steps; then compare ego-motion with all vectors of corresponding features to remove motion of ground features. The remained features are shown on display to warn the driver of potential obstacle.

The remainder sections are arranged as follows. In Section II, we describe the Ego-motion estimation. The method of object detection is described in Section III. Experiments and results are presented in Section IV, with conclusions and future works in Section V.

## II. EGO-MOTION ESTIMATION

Prior to detection, we use the input images to estimate the ego-motion of the vehicle. The estimated ego-motion is used

during detection to remove ground level features that might be wrong detected as obstacles.

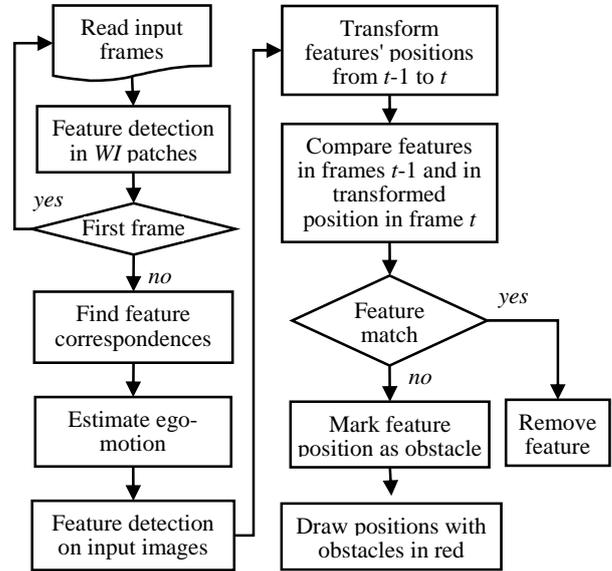


FIGURE II. OBSTACLE DETECTION PROCEDURE.

### A. WI Patch

We define a region around the vehicle which we named "WI patch", WI stands for world space to image space, because it is a data structure storing mapping from WCS to ICS. The WI patches are two 0.3m by 1.5m regions on both sides of the vehicle with x-axis crossing its middle as shown in Fig.3. It is the region we use to estimate the ego-motion of the vehicle. The dimension is chosen from the maximum displacement of a regular vehicle during a single frame. Because majority of countries have a top speed limit of 130km/h [11], which is 1.2037m/frame, we choose 1.5m so that enough features can be matched between two frames when the vehicle is moving within the speed limit. The 0.3m width is calculated from the turning radius of smallest, widely available production car currently on market, Smart Fortwo™, which is 4.37m [12]. At 130km/h, the vehicle would move about 0.15m sideways per frame at 30fps, thus the 0.3m width for the patch. The resolution which each element of WI patch is sampled depends on the resolution of the camera and mounting position. The distance is chosen by the minimum distance of neighboring pixels transformed to x-y plane of world space.

We make the assumption that it is unlikely for above-ground objects to enter the WI patches for that it is very close to the vehicle. If a moving object enters at speed, it will most likely collide with the vehicle without possibility of avoidance.

### B. Feature Detection

For stable ego-motion estimation, we need features that have good repeatability between frames. Following methods were explored. All process works on the Y component of camera's YCbCr color space.

1) *Fast corner detection*: Features from accelerated segment test (FAST) [13] is a fast corner detection method frequently used in real-time applications.

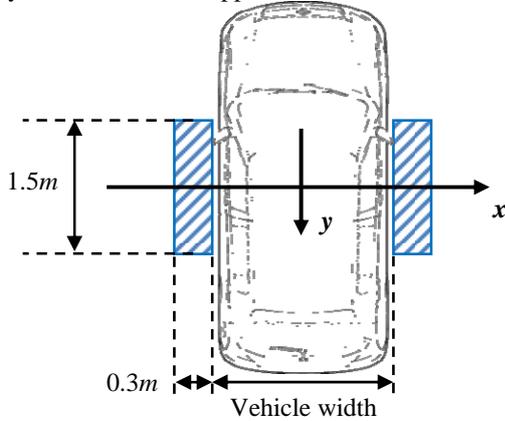


FIGURE III. POSITIONS OF WI PATCHES ON BOTH SIDES OF VEHICLE.

2) *Brightness*: From the videos recorded in our experiments, we've observed that in asphalt and concrete, which have relatively low feature count, the brightest and darkest points remains to be the brightest and darkest as local extremum. So a feature selection is devised to scan through WI patches and use the 10 brightest and 10 darkest points as features.

3) *Bi-level difference of Gaussians*: A bi-level difference of Gaussians (DoG) method used in CenSurE [14] was tried to obtain features that are less susceptible to aperture problem.

In order to have sufficient number of features for ego-motion estimation, the threshold used to determine if the pixel is a feature is adjusted by

$$T_{t+1} = \begin{cases} 1.25 T_t, & \text{if feature count} > 2L \\ 0.75 T_t, & \text{if feature count} < L/2 \\ T_t, & \text{otherwise} \end{cases} \quad (1)$$

where  $T_t$  is current threshold,  $T_{t+1}$  is the threshold for next frame,  $L$  is a constant defining the target feature count. We define  $L = 20$  by experiment, as higher values do not yield better result, and lower resulting little to none correspondences. The amount of adjustment is set to 1.25 and 0.75 for that adjustment by power of twos can be done more efficiently, and these values would adjust the threshold quickly without changing feature count erratically.

Our test shows that although *FAST* and brightness method are faster than Bi-level difference of Gaussians, their results are quite random between frames. *FAST* features randomly appear and disappear in large numbers as threshold is adjusted frame by frame in low contrast asphalt-surface frames. Brightness method initially looked promising to human eye, but test shows that it could not cope with tiled surfaces where contrast and feature count are high. The test led us in choosing bi-level difference of Gaussians, which is more stable compared to the

other two methods. **Table 1** compares the advantages and drawbacks of the three methods.

TABLE I. FEATURE DETECTION METHOD COMPARISON

Method	Advantages	Drawbacks
<i>FAST</i>	Fast.	Features vary when threshold is adjusted between low contrast frames.
Brightness	Faster than <i>FAST</i> . Work well in very low contrast.	Unstable when contrast is high. Features clustered together when large feature appears.
Bi-level <i>DoG</i>	Blob features less prone to aperture problem	Slower than the other two methods

### C. Find Motion Vectors

For all the features detected in current frame and previous frame, we perform sum of absolute difference with a  $5 \times 5$  window on all pairs to find the correspondence features, the vector  $\mathbf{V} = [V_x \ V_y]^T$  from the corresponding feature pair is the motion vector for the feature pair. Because not all features have correspondences, we define following rules used to remove outlier vectors.

*Rule 1.*  $|\mathbf{V}| > 1.2037m$ , which represents speed larger than 130km/h.

*Rule 2.*  $V_x > V_y$ , meaning vehicle is moving sideways.

Motion vectors then are sorted by length and cropped to leave 50 vectors in the middle if there are more than 50 vectors. The cropping is done to reduce calculation.

### D. Least Squares Ego-motion Estimation

The ego-motion of the vehicle is

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a & b \\ -b & a \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix}, \quad (2)$$

where  $a$  is  $\cos \theta$ ,  $b$  is  $\sin \theta$ ,  $\theta$  is the turning angle,  $[x \ y]^T$  is feature's world coordinate in current frame,  $[x' \ y']^T$  is the feature's world coordinate in previous frame, and  $[t_x \ t_y]^T$  is the displacement vector. Ego-motion is estimated using least squares method by rewriting Eq.(2) and combine all feature pairs into

$$\begin{bmatrix} x_1 & y_1 & 1 & 0 \\ y_1 & -x_1 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_n & y_n & 1 & 0 \\ y_n & -x_n & 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ t_x \\ t_y \end{bmatrix} = \begin{bmatrix} x'_1 \\ y'_1 \\ \vdots \\ x'_n \\ y'_n \end{bmatrix}, \quad (3)$$

where  $[x'_i \ y'_i]^T$  are the features from previous frame and  $[x_i \ y_i]^T$  are the features from current frame. The resulting ego-motion is the movement of the center of the vehicle of previous frame which is located at origin of world coordinate system.

### E. Dampening Ego-motion

The estimated ego-motion displacement  $\mathbf{t} = [t_x \ t_y]^T$  is dampened by

$$\mathbf{t}_{cur} = \frac{1}{2}(\mathbf{t} + \mathbf{t}_{prev}), \quad (4)$$

where  $\mathbf{t}_{prev}$  is the ego-motion displacement vector of previous frame and  $\mathbf{t}_{cur}$  is the ego-motion of current frame to reduce the error caused by mis-corresponded feature points.

## III. DETECTION

The detection is done by comparing feature points in previous frame and in current frame displaced by ego-motion. This section describes the steps in detecting obstacles and eliminated ground-surface features such as lane markings that are not obstacles. By estimating ego-motion accurately, above-ground objects with detectable features will be detected, surface features will be rejected.

### A. Feature Detection

Every single pixel from input images that can be transformed into  $x$ - $y$  plane of world space can be used as a feature for detection, but it would be very time consuming. Thus we use *FAST* to detect features from all the input images.

### B. Remove Surface Features

All features on ground surface, such as lane markings and manholes, will always have a rigid motion on hard ground. The ground's motion is the opposite of ego-motion, so transform a ground feature  $F_t$  in current frame by ego-motion, we get its position that feature  $F'_{t-1}$  in previous frame. As shown in **Fig.4**, the feature  $F'_t$  in *WCS* is calculated from  $F_t$  in *ICS* by inverses of camera transformation and distortion function and  $F_{t-1}$  is calculated by projecting  $F'_{t-1}$  back to *ICS*.

We define a threshold  $T_g$  that  $F_t$  and  $F_{t-1}$  are the same surface features in current and previous frame if the sum of absolute difference is less than  $T_g$ , otherwise  $F_t$  is part of an obstacle.

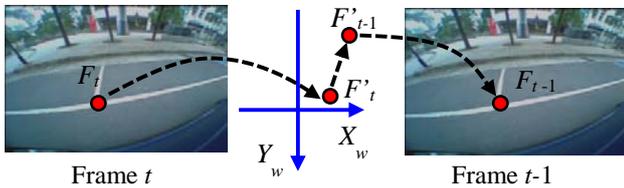


FIGURE IV. TRANSFORM FEATURE  $F_t$  IN CURRENT FRAME TO POSITION IN PREVIOUS FRAME.

## IV. EXPERIMENTS AND RESULTS

Experiments using actual vehicle was performed. The experiments were set to test the detection performances on different types of road surfaces. Tests were also performed at

different speeds to see how speed affects the accuracy of detection process.

In the experiment, the test vehicle is a 2007 Suzuki Swift 1.5L *GLX*. The dimension of the vehicle is 3.755m long and 1.69m wide. The vehicle has a small 4.6m turning radius, which is useful when testing ego-motion while performing turns on road-side parking scenario.

Four 137° field-of-view cameras are mounted at each side of the vehicle. Front camera is mounted above the number plate; rear camera is mounted on the flat surface above the rear bumper; and side cameras are mounted under the side mirrors. Such setup is similar to various vehicle models with surround view technologies that are on the market.

All cameras are connected to an industrial computer with a 4-input video capture card located at rear seat. The entire system is powered with a 12VDC lead-acid battery.

During the experiments, we focused mainly on removing ground features which are often misidentified as obstacles. And because of wide-scoped view of our system, the speed which the system can operate is also investigated. The results are analyzed by vehicle speed and road surface types.

### A. Surface Feature Removal

The types of road surfaces we tested are asphalt, concrete, tiled cinder blocks. The accuracy of surface feature removal on these road surface types are shown in **Table 2**. Accuracy is described as percentage of frames with full removal of surface features over each total frames.

TABLE II. SURFACE FEATURE REMOVAL ACCURACY

Road surface type	Full removal frame count	Accuracy (%)	Total frame count
Asphalt with lane markings	1,237	66.79%	1,852
Concrete	682	93.68%	728
Tiled cinder blocks	2,110	82.68%	2,552

### B. Factors Affecting Surface Feature Removal

Upon frame-by-frame inspection of our test videos, we have observed some factors that result in the failure in ground feature removal.

1) *Error in estimated parameters*: The errors in estimated parameters cause the pixels near the edge of image being transformed to incorrect positions when transformed from *ICS* to *WCS* and back to *ICS*.

2) *Error in ego-motion estimation*: The error in rotation in ego-motion estimation causes large displacement at distance. Although the error would only cause erroneous detection at distance which can be safely discarded, the error would produce more obstacle features which affect overall run time. Another problem related to this error arises when we try to limit number of detected obstacle features by adjusting threshold used in surface feature removal. Increased obstacle

feature count heightens the threshold, resulting more obstacle features with less contrast getting removed.

### C. Effectiveness at Higher Speed

In order to cover entire surrounding area of the vehicle, wide-angled cameras are used. Wide-angle implies the area encompassed by a pixel increases rapidly with increasing distance. At 50m, the pixel is over 0.3m wide, where smaller obstacles such as barrier posts and small animals are difficult to detect. The limit on detection range reduces the effective speed which it can be operated.

### D. Ego-motion Accuracy

By inspecting frames with high amount of falsely identified objects together with its detected ego-motion, it shows that ego-motion accuracy is the main contributing factor to the accuracy of detection. As shown in **Fig.5**, the correct ego-motion would eliminate all surface features when ego-motion is correctly estimated.

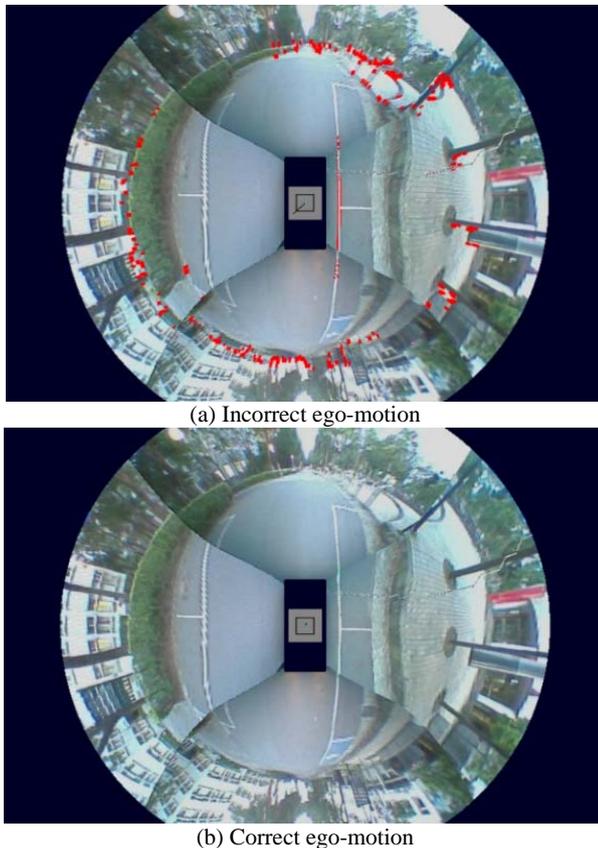


FIGURE 5. DETECTION WITH INCORRECT EGO-MOTION AND CORRECT EGO-MOTION.

## V. CONCLUSIONS

Compared to current surround view systems, the proposed system gives the driver immediate view of vehicle's surrounding environment and the freedom to change view point with current region of interest. The visual detection system

provides additional hint of above-ground objects in the environment, allow faster focus on potential obstacles. The direct projection of input image onto the dual-camber model also increases the quality of output imagery with less memory requirement. The visual method used for detection removes the need for additional sensors, reduces the system cost. The reduced cost is transferred onto the processing power of the processor, which can be more easily integrated into vehicle's other systems, leading to better integration.

The system developed produces promising results on vision-based detection for vehicles. It also demonstrated a cost-effective method into building a vehicle safety system based on low-cost cameras with no radar and ultrasonic sensors. With recent advances in vehicle electronics such as NVIDIA's Jetson platform [15], our detection system can be integrated with it, taking advantage of massively parallel computation hardware. Modifying ego-motion estimation and detection to use parallel platforms *CUDA* or *OpenCL* would improve the performance by several times. Once integrated with vehicle's on-board computer, standard on-board information like speed sensor can also improve the ego-motion estimation, providing better detection result.

## ACKNOWLEDGMENTS

This work was supported in part by the Ministry of Science and Technology, Taiwan under the grant of the research project MOST 106-2221-E-008-097-MY2.

## REFERENCES

- [1] Nissan, Around View Monitor, <https://www.nissan-global.com/EN/TECHNOLOGY/OVERVIEW/avm.html>
- [2] Honda, Multi-view Camera System, <http://world.honda.com/news/2008/4080918b-eng.html>
- [3] T. Ehlgren, M. Thom, and M. Glaser, "Omnidirectional cameras as backing-up aid," in Proc. of IEEE 11th Int. Conf. on Computer Vision, Rio de Janeiro, Brazil, Oct.14-21, 2007, pp.1-5.
- [4] T. Gandhi and M. M. Trivedi, "Vehicle surround capture: survey of techniques and a novel omni-video-based approach for dynamic panoramic surround maps," IEEE Trans. on Intelligent Transport Systems, vol.7, no.3, pp.293-308, Sep. 2006.
- [5] M. Bertozzi, A. Broggi, P. Medici, and P. P. Porta, "Stereo vision-based start-inhibit for heavy goods vehicles," in Proc. of IEEE Intelligent Vehicles Symp., Tokyo, Japan, June 13-15, 2006, pp.350-355.
- [6] Y.-C. Kuo and D.-C. Tseng, "Wide-scoped around view monitor," *Int. Journal of Advances in Electronics and Computer Science*, vol.4, is.11, pp.82-88, Nov. 2017.
- [7] D. Hoiem, A. A. Efros, and M. Hebert, "Putting objects in perspective," *Int. Journal of Computer Vision*, vol.80, no.1, pp.3-15, Oct. 2008.
- [8] A. Saxena, S. H. Chung, and A. Y. Ng, "3-D depth reconstruction from a single still image," *Int. Journal of Computer Vision*, vol.76, no.1, pp.53-69, Jan. 2008.
- [9] W. Enkelmann, "Obstacle detection by evaluation of optical flow fields from image sequences," in Proc. of 1st European Conf. on Computer Vision, Antibes, France, Apr.23-27, 1990, pp.134-138.
- [10] T. Gandhi and M. Trivedi, "Parametric ego-motion estimation for vehicle surround analysis using an omnidirectional camera," *Machine Vision and Applications*, vol.16, no.2, pp.85-95, Feb. 2005.
- [11] Wikipedia, "Speed limits by country", [http://en.wikipedia.org/wiki/Speed\\_limits\\_by\\_country](http://en.wikipedia.org/wiki/Speed_limits_by_country)

- [12] Smart, "Intelligent electric mobility",  
<https://www.smart.com/en/en/index.html>
- [13] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in Proc. of European Conf. on Computer Vision, Graz, Austria, May 7-13, 2006, pp.430-443.
- [14] M. Agrawal, K. Konolige, and M. R. Blas, "CenSurE: Center surround extremas for realtime feature detection and matching," in Proc. of 10th European Conf. on Computer Vision, Marseille, France, Oct.12-18, 2008, pp.102-115.
- [15] NVIDIA, "Jetson automotive development platform,"  
<http://www.nvidia.com/object/jetson-automotive-developement-platform.html>