2014 13th International Conference on Control, Automation, Robotics & Vision
Marina Bay Sands, Singapore, 10-12th December 2014 (ICARCV 2014)

Traffic Cone Detection and Localization in TechX Challenge 2013

Lubing Zhou
Institute for Infocomm Research
A^*STAR, 1 Fusionopolis Way
#21-01 Connexis (South Tower)
Singapore 138632
Email: zhoulb@i2r.a-star.edu.sg

Han Wang, Danwei Wang
and Lihua Xie
School of Electrical and
Electronic Engineering
Nanyang Technological University
50 Nanyang Ave, Singapore 639798

Keng Peng Tee
Institute for Infocomm Research
A^*STAR, 1 Fusionopolis Way
#21-01 Connexis (South Tower)
Singapore 138632

***Abstract*—This paper presents the detection and localization methods of entrance and staircase markers for the team *E-Mobile* in TechX Challenge 2013. Autonomous vehicles are required to detect and locate traffic cones beside the indoor entrance and staircase. One big challenge is from the unpredictable lighting conditions and environment. Different practical techniques such as color space selection, segmentation, shape analysis, distance estimation, and detector training are combined to obtain good detection rate and localization accuracy. The proposed methods can achieve satisfactory performance in real-world experiments.**

***Keywords*—object detection, object localization, autonomous vehicle.**

I. Introduction

Autonomous intelligent systems have attracted great research interests in recent years, especially for autonomous vehicles [1], [2]. Vision module is one key component in TechX Challenge 2013 autonomous robot competition [1]. The ultimate mission is to search specified targets including water bottle, mannequin and dustbin in a realistic big urban area, which covers over 200 meters by 200 meters comprising of many indoor zones as well as outdoor zones. The robots are expected to search three types of targets: mannequin, trash bin and water bottle. To achieve such ultimate goal, there are also many necessary preceding components such as global navigation and localization, local indoor and outdoor exploration, indoor entrance and exit negotiation, staircase climbing. One practical task is to detect and locate traffic cones beside the door and staircase as displayed in Fig. 1.

Although traffic cones appear to be easily distinguishable, there are still many challenges for this practical application. First, the robots run in uncontrolled urban environment, which requires the vision capability to handle various scenarios such as indoor, outdoor, sunny, cloudy, backlighting, frontlighting, clutter and shadow of trees. Second, due to the exploration strategy, detection of traffic cones allows zero tolerance of errors. The robot is expected to be navigated to a GPS position near the entrance or staircase, and then taken over by vision system to guide it to the front of entrance or staircase. If there are any false detections, the robot will fail because no other information can be relied on. Sometimes, the viewing distance and angle vary greatly depending on the GPS accuracy. In addition, real-time performance is also a challenging requirement. Taking entrance markers as example, the following several sections will introduce a fusion method to detect and locate the traffic cones.

[1] http://www.dsta.gov.sg/techx-challenge

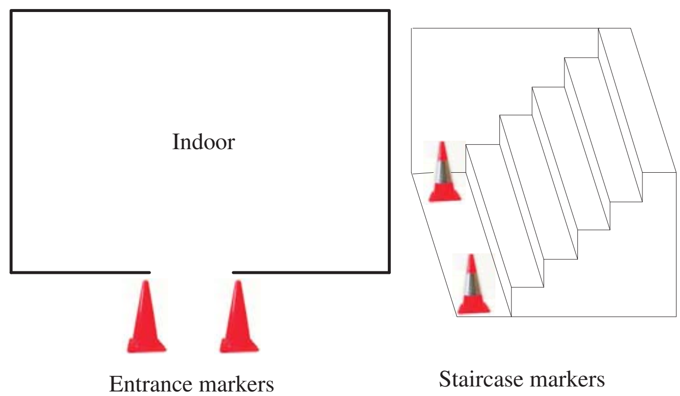

Fig. 1. Competition scenarios of entrance and staircase markers, the two cones are about 1 meter apart.

II. Color Model Selection

Selecting a proper color model is critical for object detection in robotic applications [3]. In computer vision, the most frequently used color spaces are intensity (I), RGB and HSV, even though there exists many other color spaces. In practice, there is no color model that can handle all cases, and different colored targets need to find specific color model to get better performance. RGB is the raw data from camera, and it changes drastically with the lighting brightness. Since the robot goes through indoor and outdoor, RGB data becomes unreliable. To reduce distortions caused by alternating lights and shadows in an image, the RGB values are normalized. For a pixel with R, G, B values, its normalized space (rgb) are calculated by:

$$r = \frac{R}{R+G+B} \tag{1}$$

$$g = \frac{G}{R+G+B} \tag{2}$$

$$b = \frac{B}{R+G+B} \tag{3}$$

Commonly, the normalized rgb color is relatively more reliable than RGB. The cone detection algorithm adopt a color space with three channels: Intensity I, 255g, 255r. The intensity component describes the overall brightness and other two components describe the percentages of green and red color. This color space is specially designed for the color of traffic cones.

III. Regions of Interest (ROIs) Extraction by Color Distance

Regions of interest are first extracted from the background based on color, which dramatically saves the computational cost on unrelated regions. One simplest way is to define a color range and verify whether the pixels fall in the range. However, such a color range is very hard to define. On one hand, if the range is defined too narrow, it will miss many authentic traffic cones under poor lighting condition. On the other hand, if the criterion is too loose, often the extracted regions of interest are not significant. The ROIs of cones can also be extracted using sophisticated methods such as salient models [4], [5].

Instead, a set of color templates $(T_1, T_2, \cdots, T_N)$ are collected for traffic cones under different lighting conditions and environments. These color templates correspond to many isolated points in the color space. Given a frame, color dissimilarity or color distance to the stored templates is calculated at every pixel. The color distance d of input color C is defined as the least distance between the input color to these color templates.

$$d = \min_{i=1}^{N} \parallel C - T_i \parallel_2 \tag{4}$$

Fig. 2 illustrates the idea to calculate the color dissimilarity. Intuitively, those pixels with similar color to traffic cones will lead to small values in the dissimilarity image. Based on this, a binary image can be obtained by applying a threshold to the dissimilarity image as shown in Fig. 3. This binary image indicates the candidate regions of the traffic cones.

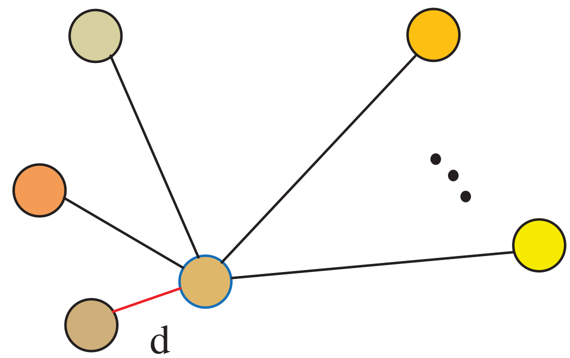

Fig. 2. Illustration of computing color dissimilarity for input color C.

IV. Contour Extraction and Verification

Before contour extraction, the binary image is successively fed into dilation and erosion morphological operations to remove small holes in the image, which makes the valid binary regions smoother. Afterwards, contours are retrieved from the binary image using the algorithm in [6]. In practice, only the external contours are extracted according to appearance of the cones.

The resulting contours usually contains many regions which are not really the cones. At early stage, the contour's area measured in pixels can be used to filter out those obviously incorrect contours. For example, a red barricade is very likely to result in a contour from previous steps, and the contour area judgement can remove it from the candidates. Specific to the entrance cones, the minimum-area bounding rectangle of the contour can be extracted. The ratio of height to width and the rotation angle of the rectangle are verified to match the traffic cones within certain tolerance. In the implementation, the ratio must exceed 1.2 and the rotation angle is below 30 degrees. As a result, fewer qualified ROIs survive after these filtering.

V. Fusion with Depth for Geometry Verification

The robots are equipped with a stereo camera (Point Grey Bumblebee XB2). Using the left and right RGB images, disparity image is often calculated by applying block matching [7] or other methods [8]. Subsequently, the depth value at each pixel can be generated by using the calibrated camera parameters:

$$depth = \frac{f * B}{disp} \tag{5}$$

where f, B and $disp$ respectively denote focal length, baseline and disparity. In the result, some pixels will produce irrational depth values originated from the calculation errors of disparity. Manually, these pixels are set to zeros or negative values, leading to many small hollows in the depth image. These erroneous pixels are masked as invalid pixels. Afterwards, 3D coordinates XYZ at each valid pixel (x, y) are calculated by:

$$X = \frac{(x_0 - x)depth}{f} \tag{6}$$

$$Y = \frac{(y_0 - y)depth}{f} \tag{7}$$

$$Z = depth \tag{8}$$

where (x_0, y_0) corresponds to origin pixel in the image corresponding to the focal center.

Using the prior knowledge, it is noticed that traffic cones are placed on the ground plane, hence we can narrow the possible regions using Y (height) and Z (depth) information. The main purpose is to prevent the false detections with obviously incorrect height or distance. As shown in Fig. 3, a

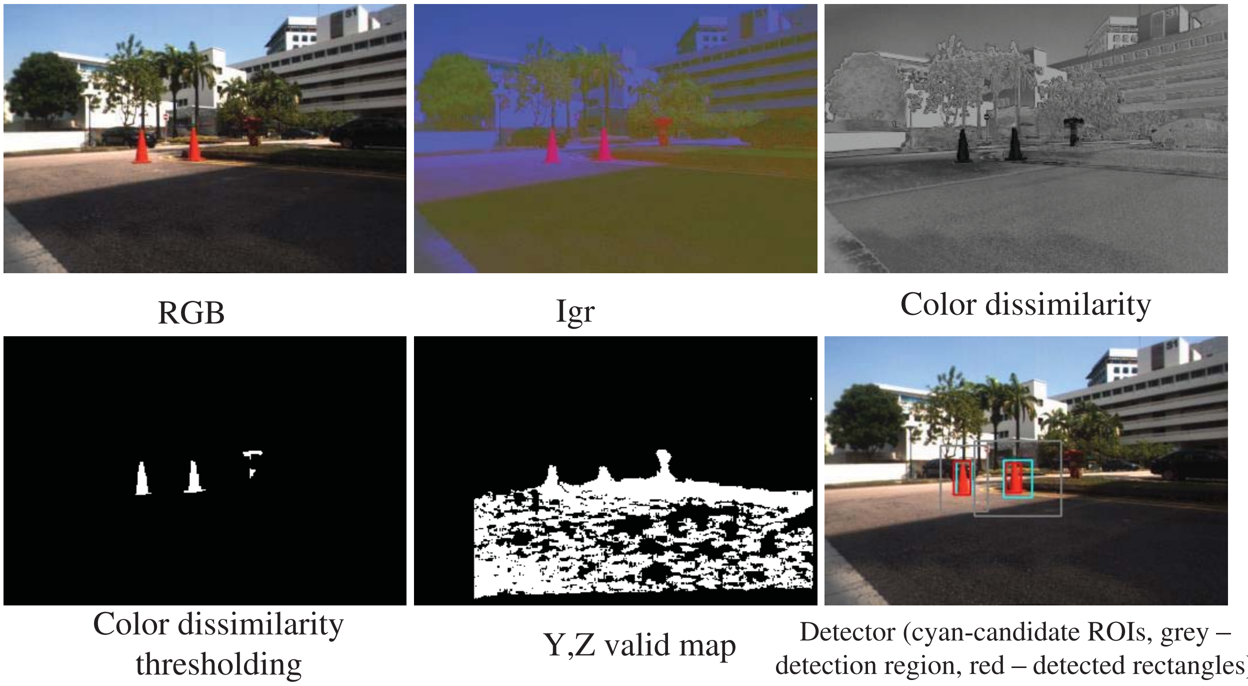

Fig. 3. This figure shows some key intermediate results during the detection process.

mask image is generated by imposing constraints $Y < 2$ meters and 1.2 meters $< Z < 10$ meters.

VI. Depth Estimation and Area Verification

There still remains a few qualified regions after applying previous filtering. One principle is to ensure all those regions with cones will pass through and meanwhile reduce fake regions as much as possible. As one example, very loose criterion of area measured in pixels is adopted as it depends on the distance between camera and cones. Since real area of traffic cones is standard, this clue will greatly enhance the detection confidence. To get the real area, the position of the object should be computed.

First, "AND" operation is applied between the binary image from color segmentation and the valid mask resulted from Y, Z verification. This will generate another mask image with many isolated regions. These regions contains the traffic cones. For each region, the rough depth is first calculated as the average of depth values of the valid pixels in the mask image. This rough depth may include the contributions from background pixels around the cones as well. Furthermore, those pixels whose depths are within half meter distance to the rough average depth are used to recalculate the depth of the object. This processing can prevent computing the depth of cone using background pixels that are far from the cone.

With the depth value, the physical areas of candidate cone regions are then computed by:

$$\text{Area} = \text{Area in pixels} * \frac{depth * depth}{f * f} \tag{9}$$

Considering the real cones, areas of qualified regions should fall into the range of 0.10 m^2 to 1.2 m^2 to allow a loose tolerance. The loose criterion again ensures all real cones can pass through to the next step. Meanwhile this further reduce the outliers in a few candidate regions.

VII. Confirmation by Detector Training

To sum up, previous steps use features such as color, shape, aspect ratio, area and geometry information to detect the cones. Fairly good performance can be obtained. However, the competition scenario allows zero false detections which would probably cause disastrous effect to the robot behavior. Moreover, it also requires the continuous coordinates of two cones to navigate the robot. The proposed method further utilizes training algorithms to detect traffic cones from the qualified regions for double confirmation. Cone detectors are learnt with collected large-scale training data. This essentially guarantees the hit rate of detections.

Basically, AdaBoost algorithm [9] is adopted to train the classifier. In principle, it combines a set of better-than-chance weak classifiers to form a strong classifier. This work uses open source OpenCV implementation [2]. The classifier utilizes the LBP features (Local Binary Patterns) [10] instead of traditional Haar features. We used 1600 samples of traffic cone images in the training, and each sample generates 4 perturbations by applying certain distortions in deviation, rotation and scaling. Hence, a total of 6400 cone images are used to learn the detector. Fig. 4 shows a number of samples.

After training, the detector is only applied to the ROIs as shown in Fig. 3 (with certain neighborhood extension). This greatly reduces the detection time compared to detect the

[2]http://opencv.org/

Fig. 4. Training samples of entrance traffic cones.

whole image. As a result, the detector will define the rectangles around the traffic cones (red rectangles in Fig. 3).

VIII. Cone Localization and Final Selection

Till now, enough cues have been extracted to get a confident detection of cones. One important thing is to accurately calculate their coordinates referring to the robot. The key point is to compute the depth values of the cones, with which their 3D coordinates can be computed accordingly. In Section VI, the depth value has already been roughly estimated in order to remove outliers using area. However, at that stage, the regions used to calculate the depth value may not be very accurate. Thus, it is necessary to recompute the depth values of cones in consideration of precision.

Depth values are calculated within the detected rectangles by the detector. In the rectangle, valid pixels are considered as these pixels whose color, Y and Z values are qualified as described in Section III and V. Similar to previous depth estimation, the average depth value of these valid pixels is calculated first, and then the portion of valid pixels whose depth values are within deviation of 0.4 meters are used again to compute a refined depth value. This refined depth is the final depth of the traffic cone. Afterwards, the 3D coordinates is then obtained using Eq. (6)-(8). Particularly, the corresponding pixel of the cone in the image is the center pixel of the detected rectangle.

Based on the cone coordinates, the final step is to select two final ones as the pair of traffic cones. Occasionally, only one cone is detected in the frame, and another cone is probably filtered out by the color or contour shape filtering. The cone detector then runs in nearby horizontal neighborhood of the already found cone, which may detect the missing cone that is need for entrance negotiation. This helps to increase the detection continuity as the navigation requires continuous input of the cone position. Sometimes, there may exist more than two detected cones because of duplicated detections of a single cone. The distance is judged to select the most suitable two cones because the distance between the pair of cones is around 1 meter.

IX. Experiments

To evaluate the performance, four video sequences are collected under real-world conditions for the entrance and staircase markers. These sequences contain complex conditions such as indoor and outdoor, sunlight, shadows and reflections. Fig. 5 displays a typical frame for each sequence. Sequence 1 records the entrance markers in the real competition scenario, and the color of background wall can easily confuse the cone detection. Sequence 2 records the outdoor staircase markers in the competition, and the difficulty here is from the extreme lighting condition at the border of shadows and sunlight. Sequence 3 records the indoor staircase markers, which is captured under biased lighting from the window. The last sequence records the entrance markers in NTU campus, and the fire hydrant and shadows are the interference factors.

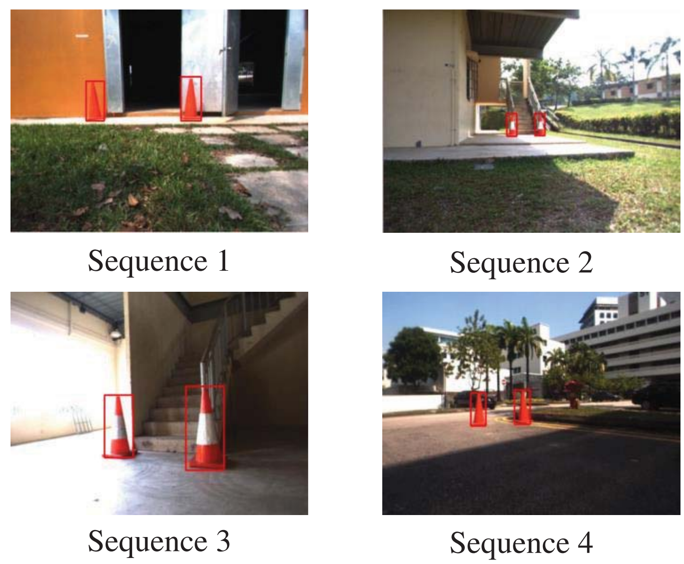

Fig. 5. Typical scenarios for the four video sequences.

Table I reports the detection results for these four sequences. It is noticed that the proposed method can achieve zero false detections while retain fairly high detection rate, Which satisfy our practical requirement well. Note that the missing rates of Sequence 1 and Sequence 2 are higher because of the confusing orange wall and harsh lighting condition between shadows and sunlight.

The algorithm is able to detect the cones up to 10 meters. The authors do not conduct thorough quantitative analysis on the localization accuracy as it is very hard to get the accurate ground truth of cone coordinates. Roughly, the localization error is within 20 centimetres, and testing experiments show the localization accuracy is adequate for the robot to pass through the two entrance or staircase cones. In other words, the localization accuracy can fulfil the competition requirements.

TABLE I
DETECTION RESULTS OF TRAFFIC CONES.

Videos	No. Cones	Correct	Missed	False
Sequence 1	1468	$\frac{1292}{88.0\%}$	$\frac{176}{12.0\%}$	0
Sequence 2	1190	$\frac{1026}{86.2\%}$	$\frac{164}{13.8\%}$	0
Sequence 3	1788	$\frac{1780}{99.6\%}$	$\frac{8}{0.4\%}$	0
Sequence 4	1100	$\frac{1062}{96.5\%}$	$\frac{38}{3.5\%}$	0

X. CONCLUSION

This paper has introduced the traffic cone detection and localization approaches that were developed for the TechX Challenge 2013. The proposed method belongs to a fusion method which has utilized various information including color, shape, geometry and classifier. A specific color model and color dissimilarity are first introduced to extract the color ROIs. Subsequently, contour, shape, area, and geometrical cues are combined to remove false detections while retain authentic cones. Off-line trained detector is then adopted to further enhance the detection confidence. Experiments on real-world conditions demonstrate the proposed method can fulfil the practical competition requirement in term of detection and localization.

REFERENCES

[1] A. Furda and L. Vlacic, *Enabling Safe Autonomous Driving in Real-world City Traffic Using Multiple Criteria Decision Making*, IEEE Intelligent Transportation Systems Magazine, vol. 3, no. 1, pp. 4-7, 2011.

[2] J. Levinson, J. Askeland, J. Becker, J. Dolson, et al., *Towards Fully Autonomous Driving: Systems and Algorithms*, IEEE Intelligent Vehicles Symposium, pp. 163-168, 2011.

[3] P.M. Caleiro, A.J. Neves and A.J. Pinho, *Color-spaces and Color Segmentation for Real-time Object Recognition in Robotic Applications*, Electrónica e Telecomunicações, vol. 4, no. 8, pp. 940-945, 2013.

[4] X. Hou, J. Harel and C. Koch, *Image Signature: Highlighting Sparse Salient Regions*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, no. 1, pp. 194-201, 2012.

[5] S. Lu, C. Tan and J.H. Lim, *Robust and Efficient Modeling from Image Co-Occurrence Histograms*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 36, no. 1, pp. 195-201, 2014.

[6] S. Suzuki and K. Abe, *Topological Structural Analysis of Digitized Binary Images by Border Following*, Computer Vision, Graphics, and Image Processing, vol. 30, no. 1, pp. 32-46, 1985.

[7] H. Hirschmuller, *Stereo Processing by Semiglobal Matching and Mutual Information*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 2, pp. 328-341, 2008.

[8] P.F. Felzenszwalb and D.P. Huttenlocher, *Efficient Belief Propagation for Early Vision*, International Journal of Computer Vision, vol. 70, no. 1, pp. 41-54, 2006.

[9] P. Viola and M. J. Jones, *Robust Real-time Face Detection*, International Journal of Computer Vision, vol. 57, no. 2, pp. 137-154, 2004.

[10] T. Ojala, M. Pietikainen and T. Maenpaa, *Multiresolution Gray-scale and Rotation Invariant Texture Classification with Local Binary Patterns*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 7, pp. 971-987, 2002.