TRAFFIC SIGN DETECTION AND POSITIONING FROM GOOGLE STREET VIEW STREAMLINES

Victor J. D. Tsai
Professor and Chair
Department of Civil Engineering
National Chung Hsing University
250 Kaokuang Road, Taichung City 40227, TAIWAN
jdtsai@nchu.edu.tw

ABSTRACT

Since Google Street View (GSV) services provide volumetric street view panoramas with associated geo-referenced information, this research aims to explore useful information from GSV streamlines. This research aims on developing an internet platform that integrates Google Maps API, HTML5 and JavaScript routines for accessing the GSV panoramas and associated orientation parameters in order to identify traffic signs and to determine their georeferenced positions. Being under development using OpenCV, OpenMP and C++, various techniques in detecting, extracting, recognizing, and positioning of traffic signs from cropped GSV streamlines along user-selected routes are primarily concerned for low-cost, volumetric and quick establishment of infrastructural database in public transportation networks towards value-added Location-Based Services.

KEYWORDS: Google Street View (GSV), Google Maps API, traffic sign detection and recognition.

INTRODUCTION

Traffic sign detection and recognition (TSDR) has drawn considerable research attention on developing intelligent transportation systems (ITS) and autonomous vehicle driving systems (AVDS) since the 1980’s. A TSDR system is normally composed in two specific phases: detection and recognition for identifying traffic signs from road images taken from the installed in-vehicle camera (Nguwi & Kouzani, 2006; Lorsakul & Suthakorn, 2007; Huang et al., 2008; Sallah et al., 2011; Escalera et al., 2011; Chen et al., 2013). In the traffic sign detection (TSD) phase, the system searches for a region of interest (ROI) that may contain traffic signs within a road image by applying color segmentation or shape classification techniques. In the traffic sign recognition (TSR) phase, the system identifies the category and content of the signs by applying neural network learning, template matching, image classification, or genetic algorithms. Many literatures gave a brief review and comparison on color-based, shape-based, and other approaches for TSDR upon the three properties of traffic signs in color, shape, and inner content (Zakir et al., 2011; Johansson, 2002; Nguwi & Kouzani, 2006; Fu & Huang, 2010; Escalera et al., 2011; Feng, 2014). In practice, both TSD and TSR phases involve intensive work on image processing and analysis of a single image/video frame captured by the in-vehicle cameras. There remain a number of important issues in handling these images in real-time environment: lighting conditions, blurring effects, non-ideal orientations, paint deterioration, shape distortions, scale variations, occlusion, and confusion with other similar man-made objects (Nguwi & Kouzani, 2006). These unavoidable issues make the development of TSDR systems an interesting and challenging topic in the computer vision community.

On the other hand, since Google launched its street view services in 2007 and opened to internet communities and user contributions (Vincent, 2007; Anguelov et al., 2010), many researches have been reported on various applications over street view images. Among these applications, Peng et al. (2010) developed a tour guiding video
from Google street view (GSV) images on intelligent mobile phones for interactive route recognition to car drivers. Similarly, Wu (2013) developed a panoramic image navigation system that integrates Google Maps routing and GSV panoramas for virtual touring on the Internet based on JavaScript and Ruby on Rails. Tsai & Chang (2012, 2013) developed an internet platform for accessing the orientation parameters of GSV panoramas in order to determine the global position of interested features by intersection from two overlapping GSV panoramas, which is suitable for volumetric data collection in establishing Location-Based Services (LBS) applications, in which the positional accuracy is not primarily concerned.

Unlike the general TSDR systems that deal with real-time images captured by the in-vehicle cameras, this research aims on developing techniques for detecting, extracting, and positioning of traffic signs from GSV streamlines along user-selected routes for low-cost, volumetric and quick establishment of infrastructural database in transportation networks. The framework and processes of the system under development is described in the remaining of this article.

**TRAFFIC SIGN DETECTION USING GSV STREAMLINES**

**GSV Streamline Cropping**

Natural color images were taken from a ring of eight cameras plus a fish-eye lens on top for producing the most popular 360° GSV panoramas (Anguelov et al., 2010). Each available GSV panorama can be requested in an HTTP URL form using Google Maps JavaScript API, along with the projection type, the geodetic position of the street view car and its moving direction with respect to the North at the time of image capture. However, the GSV panorama may include severe distortions and ghost image in the upper portions (image from the 9th camera shooting the sky) and lower portions (rendered from other panoramas) as shown in Figure 1(a). Meanwhile, the tiled panorama accessed from GSV server is not truly 360° surrounded, but with overlapped portion in both ends in the horizontal direction. Therefore, the overlapped portion and the empty block in the last row of image tiles will be cropped as shown in Figure 1(b). Then, it would be good to locate ROIs on the central portions of ±30° in pitch (vertical angle), as shown in Figure 1(c), according to the vertical field of view of the camera lens (Tsai & Chang, 2012, 2013).

The Google Directions API makes it easy to plan a route on the Google Maps from selected start location to end location, returning in a JSON `routes` array with intermediate `legs` and `steps` or in an XML `<DirectionsResponse>` element. These `steps` and `legs` can be used to access available GSV streamlines along the route from which intermediate locations were interpolated from the connecting `steps` within a `legs` segment. The GSV panorama and its associated geodetic position and heading information were requested from the GSV server. The panoramas were then cropped for finding ROIs in the following steps that incorporate routines being developed using OpenCV, OpenMP, and C++.

**Color Segmentation**

Color tone is the most important and significant descriptor that simplifies and dominates feature identification and extraction in visual interpretation applications using color images. Basically, objects reflect the chromatic electro-magnetic *radiance* (EMR) and show various colors subjective to their reflectance to the three primary wavelengths in red (R), green (G), and blue (B) as designated by the CIE (Commission Internationale de l’Eclairage). Thus, all colors are seen as variable combination of the three primaries in the RGB color space, which is usually used in representing and displaying images. Color segmentation is the most common method applied for the initial detection of red, green, and blue traffic signs with white and black inner contents (Sallah et al., 2011; Benallal & Menuier, 2003). Transformations from RGB color space into many other color spaces like HSI, HSL, HSV/HSB, IHLS, YUV, YC\textsubscript{b}C\textsubscript{r}, CIELAB, CIECAM, and CIELUV have been adopted in the literatures in color-based TSD. The thresholding segmentation approach was then applied to classify pixels as traffic sign or background (non-traffic-sign) for generating a binary image.
The HSI color model (Gonzalez & Woods, 1992) is adopted in this research for manipulating images with the transformation from the RGB space in the following relations:

\[ I = \frac{1}{3} (R + G + B) \]  \hspace{1cm} (1)
\[ S = 1 - \frac{\min(R, G, B)}{I} \] (2),

\[ H = \begin{cases} 
\theta & \text{if } B \leq G \\
360^\circ - \theta & \text{otherwise} 
\end{cases} \] (3),

where \( \theta = \cos^{-1}\left( \frac{0.5[(R-G) + (R-B)]}{\sqrt{(R-G)^2 + (R-B)(G-B)}} \right) \) (4).

Transformed \( I, S, H \) components are then segmented with certain threshold values for traffic sign in different colors, resulting in a binary image in which traffic signs object pixels are represented by 1’s, and the background pixels by 0’s. The resulted binary image was then analyzed and labeled for connected regions by connecting 8-connected pixels together. The bounding box of each connected components is set to extract the candidate ROI for filling holes within its internal boundary.

**Shape Classification**

Shape is also an important attribute of the traffic signs. Depending on the regulations, traffic signs are usually categorized as prohibitory or restrictive, mandatory, regulatory, and warning signs with regular shapes of circle, triangle, diamond, square or rectangle, and hexagon. Shape classification of each candidate ROI is carried out by evaluating the ROI’s *Extent*, which is a ratio of the pixels in the ROI to the pixels of its bounding box (MathWorks, 2013), for candidate traffic signs in different shape patterns by setting appropriate thresholds (Sallah, 2011).

**ROI Extraction and Normalization**

Once a region was classified as traffic sign candidate, the image block within the bounding box of the ROI will be extracted from the cropped GSV panorama. Meanwhile, the background image will be filtered using the area within the internal boundary as a mask. Then the ROI image was normalized into a fixed dimension of 30 x 30 pixels by applying a homographic transformation in the form of homogeneous coordinates as following

\[
\begin{bmatrix}
u' \\ v' \\ \omega'
\end{bmatrix} = \begin{bmatrix}
h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & 1
\end{bmatrix} \begin{bmatrix}
u \\ v \\ 1
\end{bmatrix}
\] (5),

where \( u' \) and \( v' \) are transformed coordinates of the pixel at \((u, v)\) of the original ROI image, \( h_{11}, h_{12}, h_{13}, h_{21}, h_{22}, h_{23}, h_{31}, h_{32} \) are coefficients of the transformation matrix.

**Sign Content Recognition**

The standard image templates of traffic signs in the Road Traffic Safety Portal Site (http://168.motc.gov.tw) of Taiwan were used as the database for recognizing the inner content of normalized image of extracted ROIs. Three approaches are evaluated to identify the traffic sign, including normalized cross-correlation (NCC) matching, multilayered feedforward back-propagation neural network (BPNN) with log-sigmoid activation function, and Hopfield neural network (HNN) with NCC matching. The database was divided into three groups: train, validation, and test set. NCC template matching is considered due to its simplicity on similarity measure, robustness to varying lighting conditions, and the ease of finding a statistical interpretation, although it may suffer from the presence of non-informative ROI pixels (Escalera et al., 2011).
Sign Positioning

Once the normalized ROI has been recognized as a traffic sign, the original ROI templates from a pair of nearby GSV panoramas will be used to compute their corresponding centroids and to detect conjugate targets. Meanwhile, the centroid of the conjugate ROI templates are also used to compute the three-dimensional position of the target traffic sign by applying intersection from the known positions of the two GSV panoramas (Tsai & Chang, 2012, 2013). The positional and non-spatial information about the identified traffic signs will be stored in XML or KML formats for import into a transportation infrastructural database for value-added Location-Based Services.

SUMMARIES

This article describes the framework and processes of an on-going research that employs traffic sign detection and recognition (TSDR) and positioning from Google Street View (GSV) streamlines. The proposed system is divided into two major components: (1) an internet platform that employs Google Maps API and Web Service, CSS and HTML5 in developing an internet platform for accessing the orientation parameters of GSV panoramas along user selected routes; (2) TSDR core routines using OpenCV, OpenMP, and C++ that deal with identifying traffic signs and determining their positions by intersection from two overlapping cropped GSV panoramas. The proposed system is under development for efficient identification of traffic signs from the cropped GSV panoramas.

REFERENCES


