AUTOMATIC ROAD FEATURE DETECTION AND CORRELATION FOR THE CORRECTION OF CONSUMER SATELLITE NAVIGATION SYSTEM MAPPING

M. Kowaliszyn*, T.P. Breckon †

*Wroclaw University of Technology, Poland, michal.kowaliszyn@student.pwr.wroc.pl
†School of Engineering, Cranfield University, UK, toby.breckon@cranfield.ac.uk

Keywords: road image analysis, traffic light detection, flyover detection, satellite navigation correction

Abstract

This paper presents a novel approach for the use of on-vehicle video analysis aimed at the verification and correction of consumer satellite navigation system mapping information. The proposed system automatically detects road and environment features (e.g. flyover bridges, road junctions, traffic lights and road signs) for real-time comparison to information available from corresponding navigation mapping. This can be used both for secondary feature-based localization of vehicle position and the verification of roadway mapping information against the true environment.

1 Introduction

In the recent decade the use of satellite navigation has become common for consumer vehicle navigation purposes. The driver, after setting the desired destination on the device, is then guided through the previously calculated route using a build-in digital map. At the present time the only fully functional Global Satellite Navigation System (GPS) has a horizontal accuracy of about 10–15 meters [18]. Even though this may seem very precise in relation to triangulation distance of the satellites, such a small variation in distance can cause confusion to the driver when navigating in a dense road environment with multiple corresponding road features.

The aim of this work is to propose a method for providing information that can be used to increase satellite navigation reliability. In order to achieve this goal we use a forward facing in-vehicle video camera. The resulting video imagery is analyzed based on identified road and environment features and this obtained information is compared in real-time with satellite navigation for correctness verification. Finally, if a mismatch is detected we propose a way to additionally correct the satellite navigation mapping indication based on the features identified from a road scene.

Related works in the area of the satellite navigation correction are dead reckoning systems and map-matching algorithms [14]. The first consists of an odometer to measure the vehicle travel distance and a gyro or compass to provide moving direction [14]. Based on these the current vehicle position can be determined given an initial known starting position. The map-matching algorithms correct the vehicle position based on vehicle trajectory and the constraint that the vehicles will be on roads for the majority of the time [14]. Prior work which is in greatest similarity to our own is presented in [9]. Here it uses a 3D inertial gyroscope, GPS and an in-vehicle camera. Based on the camera imagery the road shape is derived and in conjunction with gyro data and GPS indications utilized to determine the exact vehicle position. In addition there is also a significant body of work within the area of roadside feature detection (discussed in Section 2). In this work, we construct the system for such satellite navigation mapping correction based on a range of road feature detection techniques from an on-board camera.

This paper is organized as follows. In Section 2 the video analysis techniques for various road features detection are introduced. These road features constitute the basis of the proposed system that is introduced in Section 3. This system comprises of three modules. In addition to the video analysis module, there is also satellite navigation part (devoted to obtain information from satellite navigation) and comparison module. The last one gathers information from first two modules and compares it. The video signal reflects the true road situation the driver is encountering and hence it is assumed to be ground truth in the comparison against the satellite navigation (in which a positional error is assumed [18]). In Section 4 the evaluation of video analysis techniques used is presented. Section 5 concludes the work and deliberates on the proposed system expected performance.

2 Video analysis

The video analysis techniques process image data provided from the in-vehicle video camera. The information extracted in this way is fundamental in terms of the latter reasoning used for vehicle localization on the roadmap. It is worth noting that in order to provide this comparison the same kind of information has to be taken from both sources (e.g. video camera and satellite navigation).

2.1 Road environment

The current state of the art of the road video analysis techniques indicates that in general this task is still highly challenging [2]. The reason behind it is that the road environment is complex, dynamic and indeterministic (due to various weather and lighting conditions). In general authors [7] distinguish the following types of roads: urban, highways, rural and off-road. Urban roads are usually crowded (there are lots of others vehicles and pedestrians). They have complex road structure with grid-like intersections. On the other hand, there are lots of road signs and traffic lights features and the roads are generally well lit under all conditions. Highways and rural roads are much less crowded and have a simpler structure. Highways are well-structured and well-denoted in terms of signs and lanes. However, rural roads can in general be neglected in terms of maintenance, be lacking in lanes and have a very inhomogeneous road surface. Off-road is considered to be completely unstructured. Here our consideration of the road environment is limited to all on-road scenarios. On the other hand, as stated in [7]: *"Roads are designed to be high contrast, predictable in layout, and governed by simple rules"*. This latter statement claims that even though the road environment appears to be challenging we should be able to successfully identify several fundamentals based on which we can interpret the overall road scene and draw conclusions from it.

For our purposes in this work we concentrate on following two primary road features: junctions and flyover bridges. The reasons for choosing these two road features are availability in both sources and usefulness in terms of the vehicle position determination. Another crucial condition is feasibility, meaning that information can be obtained in a relatively, readily manner. This condition is likely to be satisfied be flyovers, but not junctions. The above mentioned issues concerning the road environments (in particular complexity and road network structure) result in the fact that the precise detection of junctions from the video imagery may by unachievable under all circumstances. To counter this we therefore use two more functionalities to provide information about junction presence - sign recognition [13] and traffic light detection [10]. Both features are relatively well-structured and can be readily detected. The only road signs which lie in area of our interest are those which denote an approaching junction (Figure 10). An interesting point to note is that the traffic lights may or may not determine a junction presence as it is also additionally used at pedestrian crossings.

Figure 1: Flyover example

The techniques used to extract these road features will be now presented in more detail in the remainder of this section. It is worth noting that since the information is aimed to be processed and utilized on-line and the techniques are then restricted to real-time operation.

2.2 Flyover detection

The first road feature to be addressed is a flyover bridge (overhead cross-over). Flyover bridges are relatively distinctive objects within the road scene. We claim that this feature is especially significant on highway roads where flyovers are often found unlike many other road features e.g. traffic light. From the literature it has been shown that the problem of flyover detection from the on-vehicle camera perspective has not been directly addressed.

Flyover bridges (Figure 1) are seen from the driver perspective as solid, horizontal objects localized several meters above the road on the position where, especially on highways and rural roads, the sky is usually seen (property *A*). They are dark from the bottom because the light does not reach their underside (property *B*). The unique property of a flyover is that the vehicle passes under it and that it takes relatively short period of time, unlike the buildings (property *C*). However, the similar and hence very confusing situation might be caused be the tree branches spread above the road (property *D*). Another challenge is to distinguish flyovers devoted to vehicles from those only for pedestrians (property *E*) which not necessarily have to be present in the satellite navigation road map.

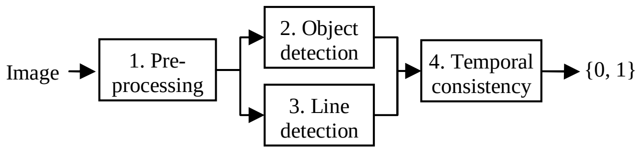
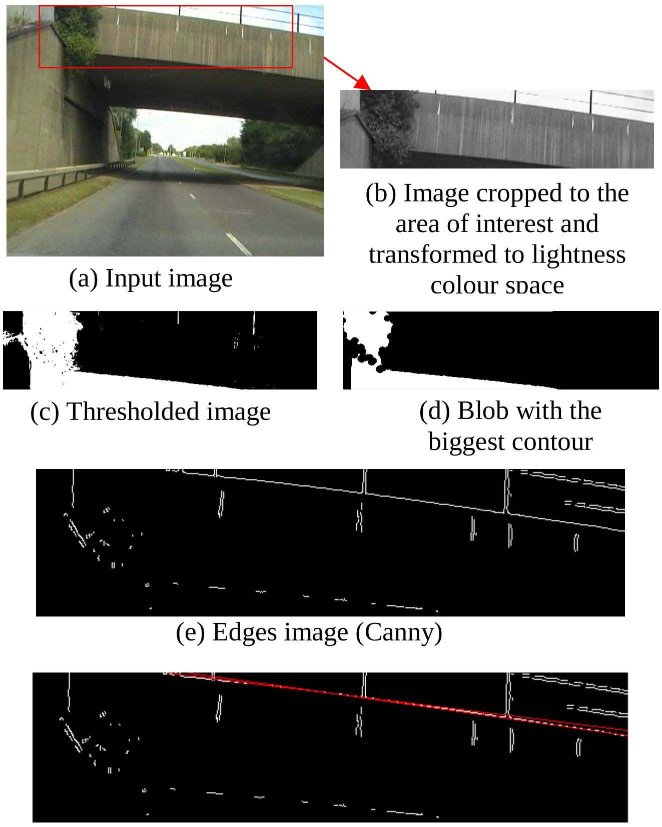


Figure 2: Flyover detection algorithm structure

Bearing in mind above flyover characteristic, the proposed algorithm of flyover detection tackles the problem twofold. It searches for the presence of the solid, dark (since properties *A*, *B*), relatively big object (property *E*) in the upper part of the image. Secondly, it requires the flyover to have horizontal property (*A*) which means that an approximately horizontal line referring to it could be localized (to deal with property *D*). If both constraints are satisfied within a given time constraints (since property *C*), then the flyover is said to be detected. The algorithm structure is presented in Figure 2 and each step shall be discussed in detail.

2.2.1 Pre-processing

As an input the video frame (Figure 3a) is taken. The first step is image pre-processing. Here the input image is cropped to the area of interest determined by the rectangle of width 80% of the input image and height 25% of it (marked with red in Figure 3a), aligned top, centre. The positional determination of this selected image subregion is based on the calibration of the camera position on the vehicle. The resulting subregion is transformed into the normalized (lightness) colour space [8] (as in Figure 3b).



(a) Input image

(b) Image cropped to the area of interest and transformed to lightness colour space

(c) Thresholded image

(d) Blob with the biggest contour

(e) Edges image (Canny)

(f) Lines detected (Hough Transform)

Figure 3: Flyover detection algorithm steps

Feature	Description	Threshold
length ratio	contour length / image perimeter	0.5
area ratio	contour area / image area	0.18
width ratio	contour width / image width	0.62
height ratio	contour height / image height	0.4
compactness	contour length / contour area	0.85

Table 1: Blob contour features

2.2.2 Object detection

This stage aims to detect large dark objects above the vehicle. The resulting image from the previous step (Figure 3b) is subsequently thresholded (with an empirical threshold value of 90 for image pixel values within the grayscale range [0, 255]). This results in the binary image (Figure 3c). A subsequent morphology operation of erosion with a rectangle-shaped kernel of size 13×13 is then performed on the image to emphasize the light regions. This favours solid objects within the scene region unlike other non-flyover occurring objects such as tree branches. From this eroded image, the connected region with the largest contour perimeter is identified (potentially referring to the flyover underside). Finally, this region is analyzed against known thresholds for size and compactness. There are 5 features taken into consideration listed and described in Table 1. All of these measures are relative measures which are independent of scale change within the image. These feature values are calculated for the identified largest perimeter region. The classification rule as follows: if the region has values above the referring thresholds (Table 1) for all 5 measures in 5 consecutive video frames then the region is classified as a flyover bridge and thus a true value output is given. Otherwise the region is rejected as a flyover bridge occurrence and a negative (false) output is given.

2.2.3 Line detection

Concurrently to the previous step line detection step is additionally performed. The idea behind this is to check if the analyzed image region contains a regular, horizontal object of significant size (as per a flyover). This is performed by the detection of a geometric horizontal line within the image.

Firstly, the edges of the cropped image region in lightness colour space (Figure 3b) are extracted. This is done using robust and commonplace Canny edge detector [3] with the empirical thresholds (200, 240) so that only strong edge gradients are returned in the resulting binary image (Figure 3e). On this image, the Hough transform [4] is employed to detect the straight geometric lines. Once again a high empirical threshold is chosen (i.e. requiring a strong evidence of the line presence) and a range of line orientation is set so that only horizontal (or quasi-horizontal) lines are detected. If any such line is detected within the range of these values a true value is returned from this component of flyover detection.

2.2.4 Temporal consistency

At final stage the data output by step 2 and 3 of this algorithm (Figure 2) is gathered and analyzed for each video frame received from the camera. The aim is to pick up only these objects which are both large, consistent and have regular shape. The classification rule is as follows: if over 15 consecutive frames both values output from Object Detection (Section 2.2.2) and Line Detection (Section 2.2.3) are consistently true, the flyover is determined to be detected within the scene. If this is so, the detection algorithm is then set to inactive for the next 15 frames of the incoming video to prevent multiple reporting of a single flyover bridge presence within the overall analysis of a video sequence for a given vehicle journey.

2.3 Junction detection

The second road feature to be discussed is a road junction. As stated in [15]: *"To date there has not been a large body of work on intersection detection"*. From the literature two approaches to junction detection were identified that could be potentially used for this purpose in our work [5, 15].

In [5], the authors use a monochromatic camera mounted on the roof of the car. This method is essentially edge-based with the main assumption being that the localisation of road edges does not change rapidly with vehicle transit. Primarily they find the junction candidate in a position where based on the information from the previous frame they would expect an absent road edge to occur. Next, the pixel values around candidate area are investigated with a region growing segmentation algorithm which is used to compare pixel values on both sides of the candidate junction road edge. The drawbacks of this method are that it cannot deal with roadside obstacles and is not resistant to any changes in a road surface (e.g. road markings or shadows). For these reasons it has a limited applicability to our requirements here. In [15], in order to be able to see both road edges, a wide-angle polycamera composed of 3 colour cameras is used. By contrast this approach to junction detection is essentially region-based. At first, the road segmentation (i.e. classification of each image pixel as road or non-road) is performed using an SVM (Support Vector Machine) [1] with a radial basis function projection as the classifier. The input feature vector comprises of 31 × 31 rectangular subimages of the road scene image. Each vector is classified as road or non-road using the previously trained SVM. Finally, based on the segmented road image, the junction is detected based on the change in shape of the road. There are 6 classes of intersections concerned: four way ("+"), 3 rotations of T-type intersection ("┬", "├", "┤") and right- ("┌") and left-angle ("┐"). The main drawback of this method is that the road surface appearance is assumed to be relatively constant *a priori* and as a consequence it cannot adapt to any road changing road surface encountered.

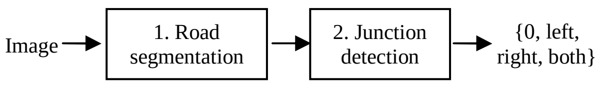


Figure 4: Road junction detection algorithm structure

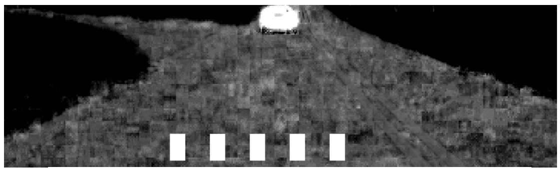
Here we aim to detect information about both the presence of a junction and about its type. We distinguish only three types of junctions: left-sided, right-sided or both-sided (four-way). We propose a technique that is similar to [15] and performs the road shape analysis based on road segmentation [2]. The principle idea is the temporal analysis of the road segmentation and to pick up junctions as instances of the road width increasing over a given temporal period. The algorithm structure is depicted in Figure 4 and shall be now discussed in further detail.

2.3.1 Road segmentation

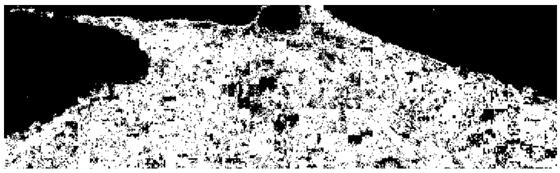
At first, the input video image (Figure 5a) is cropped to the bottom half subregion where the road area is located (Figure 5b). Next road segmentation with the method presented in [2] is performed on this subregion. The employed technique is robust to lighting variations within the road scene and can operate on various road shapes. It transforms the input image to the illuminant-invariant colour space (Figure 5c) introduced in [6]. The resulting one-channel image is then segmented using histogram model-based seeded region growing algorithm. The seeds (presented as white rectangles in Figure 5c) are located on the very bottom of the image where the road surface is assumed to be found. As a result the binary image is obtained (Figure 5d) where white pixels refer to road and black to non-road.

(a) Input image

(b) Bottom half of input image



(c) Illuminant-invariant colour space



(d) Resulting road segmentation image

Figure 5: Road segmentation algorithm steps

2.3.2 Junction detection

In order to perform junction detection we propose a simple method of road boundary analysis performed on the segmented road image (Section 2.3.1). This solution is addressed to the simple, fork–type junctions (e.g. Figure 5a) which are characteristic of rural roads and highways.

The algorithm is as follows. The input segmented road image (Figure 5d) is transformed with a morphological closing operation [17] using a circular 5×5 kernel. This transformation results in the imperfections of the road segmentation step (e.g. Figure 5d) caused by various forms of noise (e.g. video compression artifacts) are attenuated (Figure 6a). After this pre-processing step, the left and right-sided

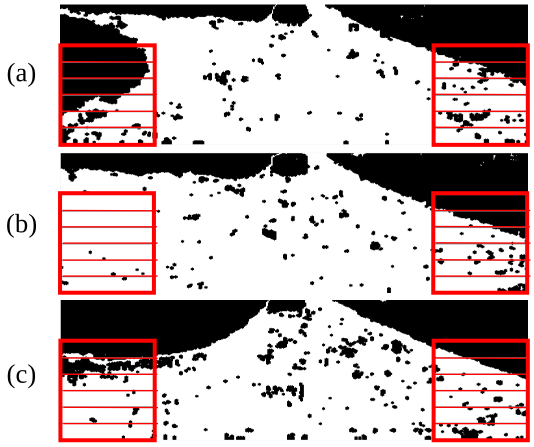


Figure 6: Junction detection process

road junction is detected by investigating both sides of the road. For this purpose we use two fixed-position rectangular windows on the each side of an image. These windows are marked with red colour in Figure 6a-c where the junction passing process is depicted. Each window consists of 6 sub-windows. We assume the junction to be present when the road gets widen for a short amount of time. The number of the road pixels is counted in each of these 6 sub-windows. If the ratio between the number of road pixels within the certain sub-window and area of this sub-window is above a threshold (empirically set to 0.25) we identify that a junction is found behind this sub-window. Finally, the overall junction classification rule is as follows: a junction is detected on the certain side only if the road surface is found behind all of the sub-windows on this side in 4–7 successive frames (these values are set for the road segmentation rate of approximately 5Hz). This enables us to avoid false alarms such as roadside obstacles (e.g. parked vehicles) and also successfully overcome potential limitations in the prior road segmentation.

2.4 Traffic light detection

Traffic lights are quite distinctive objects by design within the overall road scene because of the highly illuminant red or green colour of lights and contrasting dark surroundings (Figure 7, Figure 9). In The United Kingdom Department for Transport the model of traffic light is as presented in Figure 7. However, this template is a common variant across Europe and in traffic systems world-wide.

The simple approach to the traffic light detection is presented in [10]. The authors of this work detect the traffic light based

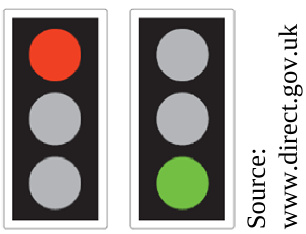


Figure 7: UK Traffic light model

on the RGB colour segmentation. More sophisticated and hence computationally intense approach is proposed in [11]. It consists of three steps: 1. detection, 2. tracking, and 3. classification. The detection is based on RGB pixel colour with temporal tracking facilitating the suppression of detection hypotheses which are not stable over multiple frames. Finally, the classification is carried out using a neural network classifier [1].

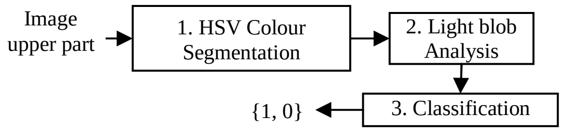


Figure 8: Traffic Light algorithm structure

As our system is aimed at real–time operation we omit the computational expense of candidates region tracking [11] and base our work on [10]. However, we add a classification step (histogram model-based) which is lacking in the original work [10]. The reason behind it is that the similar lights appear elsewhere in the road scene (e.g. stop lights on the rear of vehicles Figure 9a). This additional classification step building on the work of [10] enables us to take into consideration the surroundings of the bright red and green regions referring to the traffic light structure (Figure 7). The proposed algorithm structure is shown in Figure 8 and shall be now discussed further.

2.4.1 HSV Colour Segmentation

From the input image (Figure 9a) the upper half of the camera image is extracted as a subregion (Figure 9b). This not only decreases the computational cost but also reduces the number of potential false positives connected with the detection of vehicle stop lights in the image foreground (e.g. compare Figure 9a with Figure 9b). Next the resulting image subregion is transformed into HSV colour space [8] and both the red and green hue regions of the image are extracted by multi-thresholding-based colour segmentation. This results in 2 binary images with marked regions of green and red colour

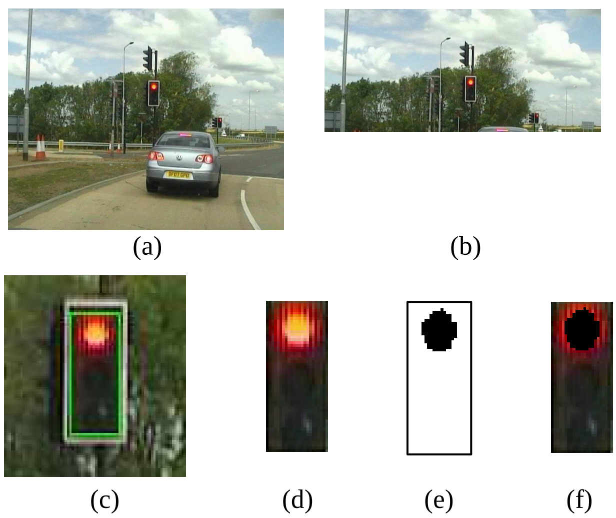


Figure 9: Traffic light detection steps

occurrence respectively (e.g. Figure 9e).

2.4.2 Light blob analysis

The resulting blobs from the detected colour lights within both images are next analyzed. Blobs which cannot correspond to traffic lights because of their size and compactness properties are filtered out. For each light blob region its size (expressed by blob area) and aspect ratio (its maximum width divided by the maximum height) are calculated. Blobs accepted as traffic lights had an aspect ratio range which was empirically set to the range between 0.5 and 2.0 and a size range between 0.009% and 0.25% of the total image subregion area.

2.4.3 Classification

In the last step we classify each blob as referring either to a traffic light or noise using a histogram model of the identified light blob surroundings. In order to do this we first extract a rectangular region of interest within the road scene image around the light blob (see green rectangle Figure 9c) in accordance with the traffic light model of the appropriate colour as presented in Figure 7. On the resulting part of the image (Figure 9d) the mask consisting of the light blob region of occurrence (Figure 9e) is applied. The resulting image (Figure 9f) contains non-light regions and constitutes a basis for a histogram-based classification. The HSV histogram of this image is compared against a collective mean histogram (which it is the normalized sum of histograms from a training set of multiple different traffic lights). The sum *s* of intersection distance [16] and correlation distance [16] between these two histograms is calculated. Finally, if the resulting sum *s* is less than a threshold value (set to 0.25) the blob is classified as a referring to traffic light, otherwise it is rejected. If any blob within the image has been positively classified, the traffic light is detected.

2.5 Road sign recognition

Here are interested in only those road signs that contain information about an approaching junction. Therefore from the complete set of UK road signs we have focused only on the types of signs presented in Figure 10. All of these signs have red rims and a triangle or inverted triangle shape but a different symbol content. Despite these obvious physical characteristics the largest challenge in the problem of sign recognition still remain as variable lighting conditions (due to weather, stroke, time of day), different states of road sign, deterioration or possible partial occlusion (e.g. by trees or other vehicles).

Figure 10: Road signs of our interest

In general, most research in the area splits the process of sign recognition into two main steps: sign detection and sign recognition. The first stage aims to detect traffic signs within the scene image whilst in the second step the meaning of the detected sign is recognized. In [12] the first stage is done by the edge extraction and then the shape analysis. However, more commonly colour scene segmentation is performed and further shape analysis on the extracted components is used [13, 19]. In the second stage, small sign candidate subregions are extracted from the image and each sign symbol is subsequently recognised. The recognition can generally be done using a machine learning approach such as probabilistic neural network [19] or via SVM [13].

For the purpose of this work we employ the technique presented in [13] and adapt it to our case (reduced set of road signs as per Figure 10). The main reason behind this choice is because this method is both rotation, translation and scale invariant and additionally robust to the aforementioned noise challenges of the generalized road sign recognition task.

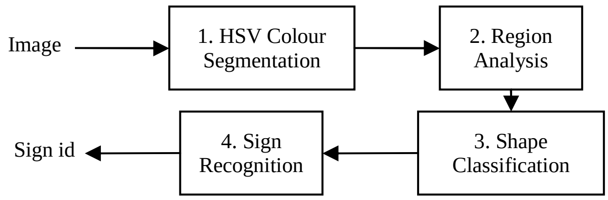


Figure 11: Sign recognition algorithm structure

The pipeline of the chosen algorithm is shown in Figure 11 and shall be now briefly discussed. Firstly the input image is transformed to HSV colour space and red colour segmentation is performed by multi-thresholding of the HSV components (Figure 11, 1). As a result, the binary image with the occurrence of the red colour areas is obtained. The resulting regions of this image are then analyzed against their size and compactness. The aspect ratio and size are calculated for each region and are accepted if they satisfy a set of *a priori* constraints (Figure 11, 2). Each region that has passed stage 2 is then classified in terms of its shape (Figure 11, 3). Two types of shapes are accepted: triangle and inverted triangle (Figure 10). As a classifier, SVMs with linear kernel [1] are used. 4. Finally, the content of each candidate sign with its appropriate symbol shape is recognized using an SVM with a Gaussian radial basis kernel function [1]. Further detail is presented in [13].

3 System design

Our aim in this work is to propose the system that can be used for a verification (and potential correction) of roadway navigation mapping information against the true environment. In order to do so we employ the in-vehicle forward facing video camera that presents the road image the driver is encountering. We perform the video imagery analysis so that

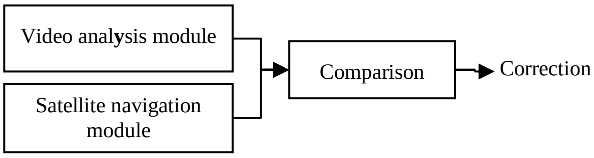


Figure 12: Proposed system structure

it can be utilized to provide a consumer navigation mapping verification. The analysis is done with regard to the road features presence which constitutes the basis for the comparison. Let us make our first assumption that the satellite navigation mapping error lies along the main road – i.e. the navigation indication is shifted against the true position along the road the vehicle is currently travelling on (assumption *A*).

3.1 System overview

A structure of the proposed system is shown in Figure 12. The video analysis module analyzes in real-time the input video imagery. In this work we decided to concentrate on the two road features: flyovers and road junctions. However, we also detect traffic light and several road signs as auxiliary features in providing information about road junctions. For the purpose of these 4 road features detection we employ the techniques presented in Sections 2.2 – 2.5 respectively. Let us denote these features as *L* – left-sided road junction, *R* – right-sided, *B* – both-sided or *J* – unspecified junction (meaning any junction) and *F* – flyover and **0** – no feature. The output of this module is the currently detected road feature (or no feature **0**) $v_0 \in \{L, R, B, J, F, \mathbf{0}\}$. Here, we additionally assume that the time of detection of a given road feature is equivalent to its ground truth occurrence at the current vehicle position (assumption *B*).

When it comes to the satellite navigation module we assume to obtain information from direct access to the satellite navigation information feed. The output of this module is the sequence $\boldsymbol{s} = (s_{-M}, s_{-M+1}, ..., s_0, ..., s_M)$ of road features (or **0**) found on the road that the vehicle is travelling on. $s_n \in \{L, R, B, F, \boldsymbol{0}\}$ and index $n \in [-M, M]$, where $M \in \boldsymbol{N}$ denotes the sequence depth (positive *n* values denote future feature occurrence, negative *n* values indicate past occurrence and $n = 0$ present occurrence). Both $v_\mathbf{0}$ and $\boldsymbol{s}$ are passed on to the comparison module which is discussed in next section.

3.2 Comparison algorithm

The comparison module links the two previously discussed modules of the proposed system. Here, the verification and then correction is carried out. In order to do this we compare the video analysis module output v_0 with satellite navigation module output sequence $\boldsymbol{s}$. We treat v_0 as reliable because it reflects the current road situation but is subject to detection noise. We also need to take into consideration that some problems will occur with road feature detection in the video analysis module.

The comparison is performed for each video frame and goes as follows: If $v_\mathbf{0} \neq \mathbf{0}$ ($v_\mathbf{0}$ refers to any road feature) then we find the element s^* of the sequence $\boldsymbol{s}$ that corresponds to the same feature ($s_n = v_0$ assuming L=J, R=J and B=J) and has the lowest absolute value of index *n* (i.e. is the closest). If there are two such elements (one future and one past) the past one is favoured. Finally, if the matching element s^* is found two operations are performed: 1. the distance expressed by the index of the picked element s^* is returned as the error and input the correction to the consumer satellite navigation, 2. the element s^* is removed from the sequence $\boldsymbol{s}$ so that it does not interfere in further matching iterations Otherwise, if any matching element has not been found, the detected road feature v_0 is treated as a noise (false detection) and ignored. In terms of satellite navigation correction it is assumed that every given instance in $\boldsymbol{s}$ and v_0 are indexed by GPS location in addition to time.

Let us now deliberate further on the assumptions made. In order to relax the assumption *A*, instead of using sequence $\boldsymbol{s}$, we should use a graph as a satellite navigation mapping representation. The graph nodes should refer to the road features and edges should determine connection between them (as the road in reality does). The value referring to an edge should contain the distance between the successive features localization. Then the video module output feature should be matched with the nearest, corresponding feature but found in the graph but not a sequence. In assumption *B* we assume the equivalence between the time of road feature detection and its physical passing by of the vehicle. However, in particular for the road signs it is not always the strictest case. In fact, the distance between the junction and the sign related to it is usually significant (approximately 10-50m) and may vary. Therefore, in practice this distance should be estimated and then output in the future. The time of transit could be approximated using either constant value or by taking into account the current vehicle velocity (time = approximate distance / vehicle velocity) which may be acquired from the satellite navigation unit. In addition, any traffic lights detected may or may not indicate a road junction as they are also used to indicate pedestrian crossings. In this case if only the junction is not present in the vicinity of detection the matching will not be disturbed (provided the satellite navigation is not erroneous at that time).

In this way using a linear matching approach a set of features v_0 extracted from the video imagery are successfully matched against a set of satellite navigation features $\boldsymbol{s}$ from which an offset either temporarily of physically can be calculated for the correction of a satellite navigation error.

4 Evaluation

In this section we present the evaluation of the video analysis techniques of a flyover detection, junction detection and traffic light detection discussed in Sections 2.2, 2.3 and 2.4 respectively and employed in the proposed system. The obtained results constitute the basis for the general system performance that we may expect.

For the purpose of testing the road video imagery from the South East of England (Bedfordshire, Buckinghamshire and Hertfordshire) was acquired. The video footage was planned to cover the different types of roads: urban (31%), rural (43%) and highways (26%) and to contain all the road features that lie in area of our interest. The road scene conditions were varied (sunny, stoke, cloudy) and all footage related to daylight driving – during the noon to late afternoon sunlight period in this geographic locale. The total test video length is 106 minutes and was recorded using interlaced DV video camera (*JVC GR-D340GK*) of power resolution 720×576 pixels. All software was written in *C++* and run on a dual-core 1.86Hz, 3GB RAM notebook.

The results obtained for traffic light method are presented in Table 2. Here by a traffic light instance we mean traffic light corresponding to each junction. However, we consider red and green colour separately as these two types of traffic light are classified using different models (collective histograms of 20 training images). As we can see in this table the method picked up 45 out of 47 traffic light instances. This gives a rate of 96% of correct detections giving only 1 false positive per 35 minutes of the footage. In the case of flyover detection method the whole video sequence contained 13 instances of flyover bridges. The proposed method detected 100% of the encountered flyovers giving only 1 false positive (caused by the tree branches spread above the road – Figure 13).

Colour	Instances	Detected	Missed	False positives
Green	29	27	2	2
Red	16	16	0	1
Both	47	45	2	3

Table 2: Traffic light detection results

In general much worst performance was obtained for the junction detection technique. For example it was tested on an 11 minute subset of the video sequence containing 11 left-sided junctions. For this test subset only 6 of them were detected with 4 false positives returns. The method seemed unable to reliably deal with right-sided junctions although further development may address this. In general, we observed 2 types of situations where the method failed. (1) When the road junction was poorly visible (e.g. occluded junctions or when the road has few lanes and the car is not travelling near the road boundary). (2) When there is a low contrast between the road and a roadside (typical for urban areas). We conclude that a polycamera (or cameras directed to the road sides) could significantly improve the detection of the road junctions [15].

The video quality (mainly due to data compression) did not facilitate proper colour segmentation for the sign recognition approach (Section 2.5). However, we can refer to the results of the authors of this method [13]. They considered 205 different Spanish traffic signs and their 5 tests covered wide range of different weather conditions. Finally, they stated: *"By inspecting the obtained results, we can say that all signs have been correctly detected in each of the five sequences at least twice"* and also: *"Experimental results indicate that our system is accurate"*. In our case we aim to recognize only 7 different types of signs (Figure 10). Therefore we conclude that the employed method is highly reliable and that given the proper video quality we can pick up the signs with the very high success rate.

Overall the success or failure of the system relies heavily on the reliable detection of road features. In instances where road features were successfully detected a viable set of video features detected v_0 could be compared and evaluated against the set of satellite feature sequences ***s.*** Clearly further work on development needs to be carried out on robust detection of road features for this to be feasible in all cases.

Figure 13: Flyover detection - false positive

5 Conclusions and further work

In this paper a proposed system for the verification and correction of consumer satellite navigation mapping is presented. It utilizes an in-vehicle video camera based on which the road scene analysis is performed. The following road features are detected: flyover bridges, road junctions, road signs and traffic light. This information is compared with a satellite navigation roadmap to verify correctness of the latter and indicate potential corrections.

An evaluation principally covered the presented video analysis techniques. The employed sign recognition method is highly reliable and accurate as well as the traffic light which success rate came out to be 96% and flyovers 100%. All these three methods give a very few false alarms. However, the results shown that the proposed junction detection technique (Section 2.3) needs to be significantly improved for a robust use. Currently it could only be realistically used as additional evidence of a left junction when the sign or traffic light has been additionally detected. Based on obtained results we may expect the proposed system to provide reliable corrections in the vicinity of flyovers and these junctions that are preceded with traffic light or road sign features.

The further research in this area will consider using additional road features (e.g. roundabouts, road markings and road curvature which is universal and can be determined everywhere). Moreover, additional inter-collaboration between the differing video analysis detection techniques could be applied so that the output of one could be utilized in the other detection approaches.

Finally, detection could also be conversely driven from the need to search for specific road features occurring within the current satellite navigation system mapping at the current reported vehicle position.

References

[1] E. Alpaydin, "Introduction to machine learning", *MIT Press,* (2004).

[2] J. M. Alvarez, A. Lopez, and R. Baldrich, "Illuminant-Invariant Model-Based Road Segmentation", *IEEE Intelligent Vehicles Symposium*, pp. 1175-1180, (2008).

[3] J. Canny, "A Computational Approach to Edge Detection", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 8, No. 3, pp. 679-698, (1986).

[4] E. R. Davies, "Machine Vision: Theory, Algorithms, Practicalities, 2nd Edition", *Academic Press*, pp. 195-210, (1997).

[5] M. Ekinci, B. Thomas, "Road Junction Recognition and Turn-Offs for Autonomous Road Vehicle Navigation", *Proceedings of the 13th International Conference on Pattern Recognition*, Vol. 3, pp. 318-322, (1996).

[6] G. D. Finlayson, S. D. Hordley, M. S. Drew, "Removing Shadows from Images", *In Proceedings of the Seventh European Conference on Computer Vision*, Vol. 4, pp. 823-834, (2002).

[7] L. Fletcher, et al., "Vision in and out of Vehicles", *IEEE Intelligent Systems,* Vol. 18, No. 3, pp. 12-17, (2003).

[8] R. C. Gonzalez, and R. E. Woods, "Digital image processing". *Prentice Hall,* (2002).

[9] Z. Hu, K. Uchimura, "Fusion of Vision, GPS and 3D Gyro Data in Solving Camera Registration Problem for Direct Visual Navigation", *International Journal of ITS Research,* Vol. 4, No.1, (2006).

[10] Y. K. Kim, K. W. Kim, and X. Yang, "Real Time Traffic Light Recognition System for Color Vision Deficiencies", *IEEE International Conference on Mechatronics and Automation*, pp. 76-81, (2007).

[11] F. Lindner, U. Kressel, and S. Kaelberer, "Robust Recognition of Traffic Signals", *IEEE Intelligent Vehicles Symposium*, pp. 49-53, (2004).

[12] G. Loy, N. Barnes. "Fast Shape-based Road Sign Detection for a Driver Assistance System", *Proceedings of International Conference on Intelligent Robots and Systems*, Vol. 1, pp. 70-75, (2004).

[13] S. Maldonado-Bascon "Road-Sign Detection and Recognition Based on Support Vector Machines" *IEEE Transactions on Intelligent Transportation Systems*, Vol. 8, No. 2, pp. 264-278, (2007).

[14] Y. Meng, et al., "A Simplified Map-Matching Algorithm for In-Vehicle Navigation Unit", *Journal of Geographic Information. Sciences,* Vol. 8, No. 1, pp. 24-30, (2002).

[15] C. Rasmussen "Road Shape Classification for Detecting and Negotiating Intersections", *IEEE Intelligent Vehicles Symposium*, pp. 422- 427, (2003).

[16] B. Schiele, J. L. Crowley, "Object recognition using multidimensional receptive field histograms", *European Conference on Computer Vision,* pp. 610-619, (1996).

[17] M. Sonka, V. Hlavac, and R. Boyle, "Image Processing, Analysis, and Machine Vision", *PWS Publishing*, (1999).

[18] C. H. Yao "Global Positioning System (GPS) Technology and Cars", (2002).

[19] K., Y. Zhang, et al. "Automatic Recognition of Traffic Signs in Natural Scene Image Based on Central Projection Transformation", *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences,* Vol. XXXVII, pp. 627-632, (2008).