Korean Journal of Remote Sensing, Vol.23, No.5, 2007, pp.365~373

Representing Navigation Information on Real-time Video in Visual Car Navigation System

In-Hak Joo †, Seung-Yong Lee, and Seong-Ik Cho

Telematics.USN Research Division, ETRI, 161 Gajeong-dong, Yuseong-gu, Daejeon 305-700, KOREA

Abstract : Car navigation system is a key application in geographic information system and telematics. A recent trend of car navigation system is using real video captured by camera equipped on the vehicle, because video has more representation power about real world than conventional map. In this paper, we suggest a visual car navigation system that visually represents route guidance. It can improve drivers' understanding about real world by capturing real-time video and displaying navigation information overlaid directly on the video. The system integrates real-time data acquisition, conventional route finding and guidance, computer vision, and augmented reality display. We also designed visual navigation controller, which controls other modules and dynamically determines visual representation methods of navigation information according to current location and driving circumstances. We briefly show implementation of the system.

Key Words : CNS, Telematics, Video, Route Guidance, AR.

1. Introduction

Recently, car navigation systems (CNS) have been developed as a key application in the geographic information system (GIS) and telematics field. Many CNSs are developed based on 2D digital map and global positioning system (GPS). Nowadays many CNS products are developed and spread to make driving easy and convenient.

However, CNSs till now have limited representation power for real world, especially in case of confusing crossroad or junction. Such defect may have drivers get lost or cause traffic accidents. In this reason, CNSs aim at more realistic representation of real world by introducing 2.5D bird-view display, and recently, elaborate 3D display features on complicated and confusing crossroads, which provides more perceptible information. The most emerging and promising trend of CNS is to directly use *real-time video* captured by camera equipped on the vehicle. The examples of such video-based CNSs are introduced by INSTAR of Siemens (Narzt *et al.*, 2003) and VICNAS of Kumamoto University (Zhencheng *et al.*, 2004). Such systems display navigation information (such as directed arrows, symbols, and icons) directly onto real video and thereby enhance drivers' understanding about real world. Such video-based CNSs are expected to reduce drivers' visual distraction while driving and to make navigation maneuver easier.

Received 27 September 2007; Accepted 6 October 2007.

† Corresponding Author: In-Hak Joo (ihjoo@etri.re.kr)

In this paper, we suggest a new CNS based on real-time video that visually represents route guidance (navigation information) overlaid with the video. Main approach of the system is to display guidance information by augmented reality (AR) scheme, supported by computer vision features. Also suggested are architecture for suggested system and visual navigation controller that determines and controls visual representation methods of navigation information dynamically according to current location and driving circumstances.

The remainder of this paper is organized as follows. Section 2 provides motive of visual navigation system, architecture of the system, and interface between modules. Section 3 describes how the visual navigation controller controls visual representation of navigation information. In section 4, we present the implementation of the system, and in section 5 we conclude our suggestions.

2. Architecture of the Visual Car Navigation System

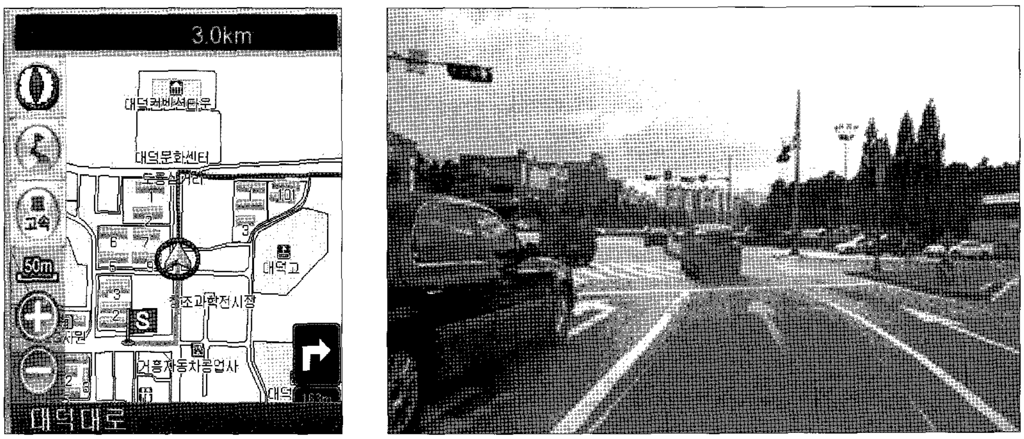
In this section, we present the motive of visual CNS and architectural framework of the suggested system. We will first discuss about the motive of visual navigation and its possible services, and present system architecture and features to implement the services.

1) Motive: Why Visual Car Navigation System?

The problem of conventional 2D map-based CNSs is that they may bring drivers' visual distraction and have drivers get lost or cause traffic accidents. To make CNS-assisted driving safer, CNSs should provide information as making the dwell time for CNS screen as short as possible. Main motive of the visual CNS is to display navigation information directly onto the real video to reduce the drivers' dwell time and visual distraction.

The representation scheme of visual CNS is essentially different from that of 2D CNS. In 2D CNS, the guidance "turn left after 300m" is represented as a highlighted line on the crossroad in 2D map, iconic or textual message on screen (Fig. 1(a)), and a voice message. In visual CNS it will be displayed on real-time video in the form of virtual (graphical) direction indicator matched with actual road in video frame (Fig. 1(b)).

In this paper, *navigation information* refers any



(a) 2D CNS (b) visual CNS

Fig. 1. Difference on route guidance display in 2D CNS and visual CNS.

information about route guidance while driving, for example, "turn left after 500m," "go straight till next guidance is notified," or "change road lane to left." Guide position means a location where the navigation information is correspond, typically street intersection, crossroad, or junction. Also, POI (Points-Of-Interests) information is also categorized into navigation information.

2) Services

Main features of our suggested video-based visual CNS are (1) to provide graphical turn guidance on reaching a guide point, and (2) to provide lane change guidance when the drivers need to change road lane in advance. The turn guidance is a basic function that all CNSs already have, while the second service is a distinctive feature of the visual CNS. The lane change guidance is realized by road lane recognition feature. Following architecture and sub-modules are designed to implement the services.

3) Architecture

To implement the services, our visual CNS introduces computer vision functions and AR display functions, as well as real-time data acquisition and conventional route guidance functions. The architectural overview of suggested visual CNS is shown in Fig. 2. The system is categorized into sub-modules: data acquisition module, navigation engine, road lane recognition module, AR rendering module, and visual navigation controller, which will be explained below. Fig. 2 also describes brief interfaces and messages between the modules.

(1) Visual navigation controller

The main part of the visual CNS is designed as a component called visual navigation controller (*VNController*). It controls other modules and flow of messages and data, as well as determines visual representation methods according to current location and circumstance. Detail method of controlling visual representation will be described in section 3. The *VNController* is also responsible for system timer control between the modules and synchronization of video and position.

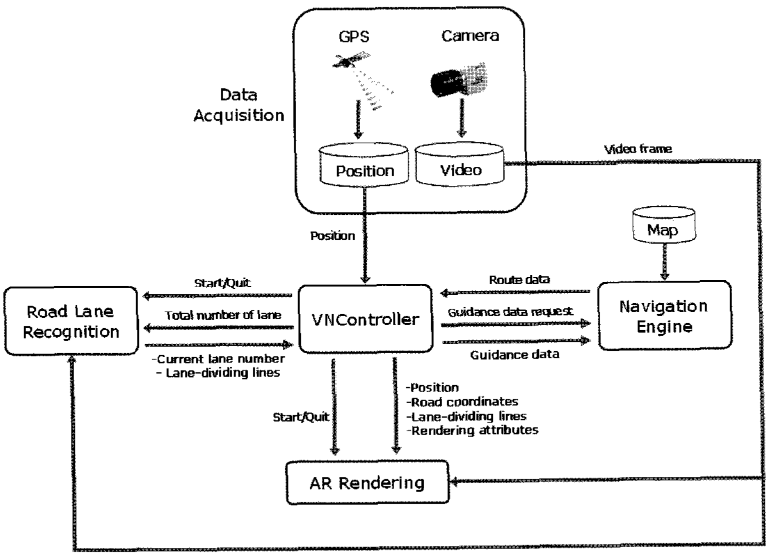


Fig. 2. Architecture of suggested visual car navigation system.

(2) Data acquisition

The position data is acquired from GPS receiver with 1-sec interval, and video frame is captured by single forward-looking camera and frame grabber connected with it. The position data includes angle and velocity data as well as WGS84 coordinates. The real-time video is sent to road lane recognition module and AR rendering module. The data acquisition module is designed to capture video into a file as well as process real-time video, in order that the video file can be also used in a simulation environment.

(3) Navigation engine

This engine includes all functions of legacy CNS such as 2D map display, route finding, map matching, and route guidance, as well as data access functions to digital map. The interaction between the navigation engine and *VNController* is designed to ensure interoperability with multiple navigation engines. Therefore third-party navigation engines can interoperate within the architecture as long as they meet the defined interface specifications. Basically, the navigation engine returns route data (a path from start point and destination point) and additional information such as turn type, path length, and estimated running time. On the mid of driving, the *VNController* compares current position and next guide point, and request additional run-time data to the navigation engine. The data are then sent to and used for *VNController*, road lane recognition module, and AR rendering module.

(4) Road lane recognition module

It executes road lane recognition process from input video. The module gets video frames, recognizes road lanes, and sends the result to *VNController*. The main objective of this module is to find current lane number on which the vehicle is going, which is one of the distinctive features of visual CNS. To support lane change service (section 2.2)), dynamic identification of current lane number is required, which cannot be obtained by conventional 2D CNS functionalities. The detailed explanation about identifying current lane number is beyond the scope of this paper, and is shown in (Holzapfel *et al.*, 2003; Lee, 2002; Yim *at al.*, 2003). The road lane recognition module returns current lane number and two lane-dividing lines represented in image coordinates to *VNController*. Because the road lane recognition process may degrade the system performance, it should be controlled to get a higher system performance. Considering the visual guidance may be unnecessary when the car is going straight without need to turn, *VNController* runs the road lane recognition module only when necessary. For example, we can control the module like as: 'start the road lane recognition 200m before the guide point, and stop it after passing guide point.'

(5) AR rendering module

This module displays graphical navigation information overlaid on the video frame. The *VNController* controls the AR rendering module by sending messages for determined visual representation method. Also sent to the AR rendering module are parameters and attribute data used for the rendering. The current location and road coordinates are sent to the rendering module for creating virtual objects (direction indicator) in 3D graphic space (world coordinate) that will be overlaid on 2D image space. The values of current location and road coordinates can determine the curvature and angle of arrow-shaped direction indicator (An example is shown in Fig. 4). The transformation between two coordinate systems is done by AR functions. Another important data for the AR rendering module is two lane-dividing lines recognized by road lane

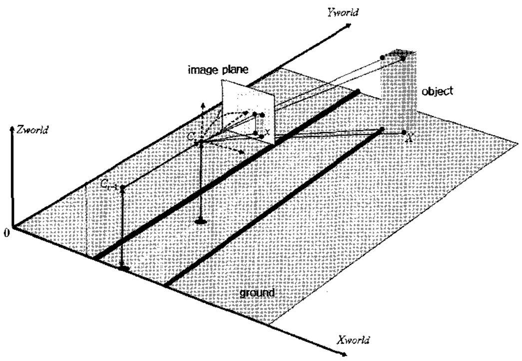


Fig. 3. Projection of virtual objects in suggested visual car navigation system.

recognition module, which is used for estimation of vehicle heading.

The virtual objects are created in world coordinates system, and will be synthesized with image by projecting them into image plane (Fig. 3). Note that, for the easy understanding, we illustrate building-shaped object instead of arrow-shaped indicator (the projection schemes are same) in Fig. 3.

In the transformation, the world coordinates are transformed to camera coordinate by

$$X_{cam} = \begin{pmatrix} R & -R\widetilde{C} \\ 0 & 1 \end{pmatrix} X \qquad (1)$$

where X is a point in world coordinate, X_{cam} is a point in camera coordinate, R is rotation parameter and $\widetilde{C}$ is translation matrix(Richard at al, 2000). Therefore, a point X in world coordinate will be projected a point x in image plane as

$$x = KR[I - \widetilde{C}]X \qquad (2)$$

For the transformation (2) we need focal length f (a component of K), rotation parameter R, and transformation matrix $\widetilde{C}$ to project world coordinate into image plane. The f is obtained by camera calibration process; R and $\widetilde{C}$ is approximated from GPS position and road lane recognition data. More detailed description is omitted here and can be found in (Hartley *at al.*, 2000).

3. Visual Representations for Navigation Information

In this section, we discuss how the navigation information is visually represented by *VNController*. The *VNController* dynamically selects visual representation methods according to current location and circumstance. For each condition, the *VNController* apply a rule to determine visual representation method. A condition can be based on distance, for example, 'how far is the next guidance point'.

1) Visual Representation Methods for Navigation Information

For implementing the visual navigation services in section 2, we designed visual representation methods for navigation information. Note that, the methods are extensible and further types can be additionally considered within the suggested architecture. The specific design for each representation method is supported by user experiments and analyses conducted for visual navigation system in (Park, 2007).

Crossroad guidance. We can represent navigation information by creating a virtual object and overlaying it on the video frame. In our system, the virtual object for crossroad guidance is designed as a curved direction indicator, whose example is shown in Fig. 4(a). To make the virtual indicator match the road in video as exact as possible, we use current position from map-matched GPS data, road coordinates from map, and vehicle heading estimated from recognized lane-dividing lines to transform the coordinates.

Lane change guidance. We represent a graphical guide for lane change when required on reaching the guide point in order not to miss the turn. Such guidance is provided for crossroads, overpasses,

(a) crossroad guidance (b) lane change guidance

Fig. 4. Visual representation methods.

underpasses, and tunnels. A pictorial example is shown in Fig. 4(b). Such type of guidance is dynamically provided, and realized by identified current lane number and turn information at next crossroad for each lane of current road. In this type of guidance, virtual object (arrow indicating lane change) is rendered onto road region of video frame.

Text and icon display. In this type of guidance, we just represent simple information such as guide type and remaining distance for the next guide point as a form of text, icon, or symbol. The display is similar to that of 2D CNS except that it is displayed on video frame. In Fig. 4, some informative icons such as current speed are also displayed on video frame.

Additional consideration about the visual representation is that it is preferable to display no navigation information in some cases. The typical case is long straight way where no guide is necessary. By pausing displaying guidance information, the system can do other process with the system resource.

2) Determination Rule of Visual Representation

This section describes how the visual representation methods above are applied according to current location and circumstance. A determination rule suggested in this paper is based on distance and current lane number, as described below. The determination rule is established in order that the methods can be applied in combination rather than they are applied exclusively. For example, the route guidance is represented as a curved direction indicator in crossroad as well as the navigation information displayed as text and symbols.

In determining visual representation, the *VNController* possibly consider that the representation in video frame is related to representation in 2D map, and that too many objects displayed may have drivers overloaded. The guideline of number of objects displayed in video is reported as 5 objects (Park *et al.*, 2007). A suggested determination rule is shown in Table 1.

In the rule, we should predefine values of parameters *distL*, *distC*, *distLE*, *distR*, that mean distance for activating mode_LaneChange before guide point, distance for activating mode_CrossRoad before guide point, distance for deactivating mode_LaneChange before guide point, and distance for no display after guide point, respectively. The values of parameters can be tuned according to road conditions, characteristics, and human factors. In our

Table 1. Determination of visual representation methods.

```
set mode_CrossRoad OFF
set mode_LaneChange OFF
set mode_TextIcon OFF
set laneChange ← 0
for each time unit, repeat
  set dist ← location~nextGuide
  set back ← location~prevGuide
  if (dist < distL)
    set mode_TextIcon ON
    set laneNum ← TotalLane (currentRoad)
    set currentLane ← LaneRecognition (laneNum)
    set tLane ← TargetLane (nextGuide)
    if (currentLane ≠ tLane) set mode_LaneChange ON
    else if (dist < distC) set mode_CrossRoad ON
  if (dist < distLE) set mode_LaneChange OFF
  if (back < distR) set mode_CrossRoad OFF
  set mode_LaneChange OFF
  set mode_TextIcon OFF
until end of video or end of route
```

system, we determine the value as *{distL=500, distC=300, distLE=200, distR=100}*. Note that, the rule in Table 1 is not uniquely established and similar variations of the rule can also be managed and applied in the same manner.

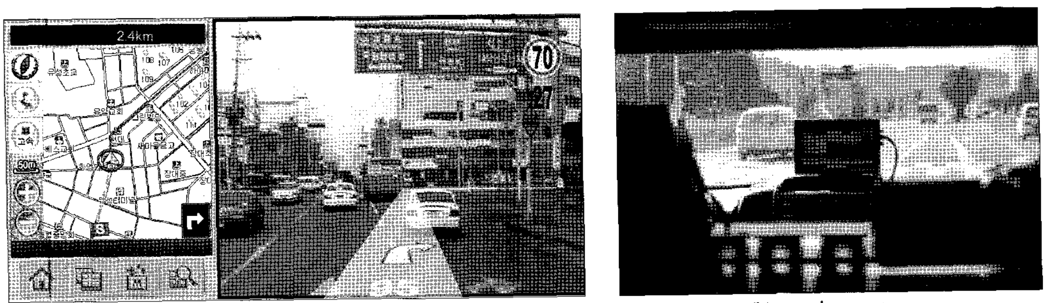
4. Implementation of a Visual Navigation System

We have been implemented a prototype visual navigation system operated on a vehicle equipped with a Pentium-4 PC, a forward-looking camera, a frame grabber, a GPS receiver, and a terminal of 7" wide screen (1024*600 resolution). Video is captured by camera in 640*480 size and 30fps, and GPS data is received in 1-sec interval.

Each module is developed as a dll or ocx with the designed interfaces implemented. The navigation engine is developed by updating and customizing that of a currently used CNS. The navigation engine also has a 2D map displayer like that of conventional 2D CNS, where the display and guidance on the 2D map are synchronized with visual representation on video by *VNController*.

Rendering module is developed as an ocx component with OpenGL functions. It plays input video on screen and displays virtual guidance objects onto the video frame. The rendering is executed by AR technology with coordinates of current position, road coordinates, and estimated vehicle heading. For directed arrows and icons/symbols that will be displayed on video frame, we designed 3D representation models and constructed them as a graphic library database.

Fig. 5(a) shows an example of execution screen of the system, where 3D virtual direction indicator representing right turn is displayed to provide visual and realistic route guidance. In this figure, mode_CrossRoad and mode_TextIcon are applied. We can see that, the turning angle of path on crossroad is used for the rendering to make the



(a) execution screen (b) running system

Fig. 5. Execution of the system.

guidance more realistic. Fig. 5(b) shows actual execution of the system operated on a PC installed on vehicle together with a GPS receiver and a USB-type camera (currently installed outside vehicle in the prototype system). As mentioned in 2.3.2, the system can be also executed in simulation mode by opening recorded file and playing it in the same manner to real-time mode.

5. Conclusion

In this paper, we suggested a new visual car navigation system that uses real video captured by camera equipped on the vehicle to represent navigation information, and introduced the implemented prototype system. By capturing real-time video and displaying graphical navigation information on it, the system can enhance drivers' understanding about route and reduce visual distraction. We designed architecture for the system and suggested kinds of visual representation methods. What we designed as a core part of the system is visual navigation controller, which dynamically determines visual representation methods for route guidance according to current location, guidance information, and road lane recognition results from the video frame.

As the future works, we are planning to implement a fully operational system that is stable for many conditions of driving. Further, the *VNController* can be extended to manage much more visual representation methods and more complicated determination rules. Also, another goal of future development is to support full compatibility with any navigation engine. The object recognition module in our system is currently developed for only road lane. But our system is expected to be extended by introducing other object recognition modules for sign lamps, road signs, and buildings, which will bring further features such as lane departure warning integrated. Finally, because the representation of visual navigation is inevitably related with human factor, issues of this visual navigation system should be approached and more studied from the view of HMI (Human-Machine Interaction).

Acknowledgements

This work was supported by the IT R&D program of MIC/IITA. [2005-S114-03, Technology Development for Construction and Management of Tangible Content for Telematics Services].

References

Hartley, R. and Zisserman, A., 2000. Multiple View Geometry in Computer Vision. Cambridge University Press.

Holzapfel, W., Sofsky, M., and Neuschaefer-Rube U., 2003. Road profile recognition for autonomous car navigation and Navstar GPS support, *IEEE Transactions on Aerospace and Electronic Systems*, 39(1): 2 -12.

Lee, J. W., 2002. A machine vision system for lane-departure detection. CVIU(86): 52-78

Narzt, W., Pomberger, G., Ferscha, A., Kolb, D., Muller, R., Wieghardt, J., Hortner, H., and Lindinger, C., 2003. Pervasive Information Acquisition for Mobile AR-Navigation Systems. In 5th IEEE Workshop on Mobile Computing Systems & Applications, Monterey, California, USA, October 2003, pp.13-20.

Park, K. S., Il-Haeng Cho, Gi-Beom Hong, Tek-Jin Nam, Jin-Young Park, Seong-Ik Cho, and In-

Hak Joo, 2007. Disposition of Information Entities and Adequate Level of Information Presentation in an In-car Augmented Reality Navigation System. Lecture Notes in Computer Science, HCI International 2007. pp. 1098-1108.

Yim, Y. U. and S.-Y. Oh, 2003. Three-feature based automatic lane detection algorithm (TFALDA) for autonomous driving, *IEEE Transactions on ITS*, 4(4): 219-225.

Zhencheng, H. and Keiichi, U., 2004. Solution of Camera Registration Problem Via 3D-2D Parameterized Model Matching for On-Road Navigation, *International Journal of Image and Graphics*, 4(1): 3-20.