International Journal of Image and Graphics
Vol. 4, No. 1 (2004) 1–18

World Scientific
www.worldscientific.com

SOLUTION OF CAMERA REGISTRATION PROBLEM VIA 3D-2D PARAMETERIZED MODEL MATCHING FOR ON-ROAD NAVIGATION

ZHENCHENG HU

Imaging Applications, Matrox Electronic Systems Ltd.
1055 St.-Regis, Dorval, Quebec H9P2T4, Canada
zhu@matrox.com

KEIICHI UCHIMURA

Computer Science Department, Kumamoto University
2-39 Kurokami, Kumamoto 860-8555, Japan
uchimura@cs.kumamoto-u.ac.jp

Received 3 June 2002
Revised 13 December 2002

This paper presents a dynamical solution of camera registration problem for on-road navigation applications via a 3D-2D parameterized model matching algorithm. The traditional camera'fs three dimensional (3D) position and pose estimation algorithms have always employed fixed and known-structure models as well as the depth information to obtain the 3D-2D correlations, which is however unavailable for on-road navigation applications since there are no fixed models in the general road scene. With the constraints of road structure and on-road navigation features, this paper presents a 2D digital road-based road shape modeling algorithm. Dynamically generated multi-lane road shape models are used to match real road scenes to estimate the camera 3D position and pose data. Our algorithms have successfully simplified the 3D-2D correlation problem to the 2D-2D road model matching on the projective image. The algorithms proposed in this paper are validated with the experimental results of real road tests under different conditions and types of road.

Keywords: Camera 3D pose estimation; registration problem; on-road navigation; parameterized model matching; augmented reality.

1. Introduction

The estimation of a moving camera's three-dimensional (3D) position and orientation are essential to the so-called registration problem in an augmented reality context. Augmented Reality (AR) supplements reality by interactively superimposing virtual objects upon the real world scene. In recent years AR techniques have been widely applied in many application areas including visualization in surgery navigation, visual reconstruction, human-computer interfaces and aviation.[1,2] In the literature of on-road navigation, the authors proposed the concept of Vision-based

Car Navigation System (VICNAS).[3] VICNAS employs image processing techniques to match the 2D digital map data with the image sequence obtained from an on-board CCD camera, and then superimposes virtual direction indicators and traffic information bulletins upon the real road scene to give drivers efficient and direct visual navigation information.

The *registration problem* is one of the basic problems currently limiting the Augmented Reality (AR) applications. The objects in the real and virtual world must properly align with respect to each other, which requires knowing the observer's exact 3D position and orientation data. Especially when the observer (camera) is moving, accurate estimation of the 3D pose data and tracking the temporal coherence from successive images will absolutely affect the synthesizing accuracy and performance of virtual objects in the AR space. The registration problem has been addressed in the early systems with some external tracking devices such as 3D gyro sensors, beacons or transponders.[2] However, the expensive maintenance cost, complicated calibration procedure and inefficient measuring ways make these devices inappropriate to most of the recent time-critical applications such as on-road navigation. With the recent advances in computing power and image interpretation algorithms, direct pose estimation from the image sequence has been extensively explored in the past decade.

The general concept of vision-based camera 3D pose estimation is to find the best set of camera position and orientation data (the six extrinsic parameters) to fit a known model in the target image. When the camera intrinsic parameters are given or available from the initial calibration, it is also known as the *absolute orientation problem*. Horn and Weldon[4] provided an initial solution of ego motion estimation with a rigid and affine model. They described the brightness constraints derived from the linear equation of intensity derivatives to rigid body motion parameters. Their solution will encounter huge blunders when the general depth information is not available. Zhuang and Haralick provided a simplified linear motion parameters estimation algorithm in their early work,[5] and it was demonstrated to be very efficient in solving the problem when there is limited noise and no corresponding errors. The algorithm is also considered to be very sensitive to noise and matching errors. In their later work,[6] iterative reweighed least squares methods were proposed as a robust solution for the affection of incorrect matching outliers, although its impractical computational cost makes it unavailable to most of the real time applications. Lowe[7] used Newton's method of linearization and iteration to perform a least-squares local minimization in his solution of viewpoint and model parameters. His algorithm extends to some more common features such as curved surface and internal parameterized models. It relies on the extraction, grouping and matching of image features to 3D models, so it is also criticized as a not very stable and problem dependent algorithm.

Our approach shares similar goals with the video-based model tracking solution described by Valinetti *et al.*[8] Valinetti introduced a scalar evaluation score

based on the local image gradient along the projected model lines to evaluate the existence possibility of certain camera pose values. His algorithm employs OpenGL's rendering engine for model projection and thus can cope with any type of object that OpenGL can render (polyhedral and smooth objects). However the estimated parameter values are still very sensitive to noise, although the visual quality of synthesizing is fairly good in the AR context.

In our work, we restrict our aim at the registration problem for on-road navigation applications. In particular, a new simplified linear algorithm is proposed to fit the 3D road shape and driving environment model to the image sequence of road scene. With the constraints of road structure and on-road vehicle motion features, this paper presents an efficient pose estimation algorithm, which converts the problem of direct 3D-2D points correlation to a parameterized road model matching on the 2D projective image. The road shape model is derived from the 2D digital road map and the absolute position data obtained from GPS and Inertial sensors. Additional road shape lookup table (RSL) concept is also presented in this paper to calculate the road model matching score. The algorithms proposed in this paper are validated with the experimental results of real road test under different road environments.

The remainder of this paper is organized as follows. Section 2 quickly reviews the system architecture of the new concept of direct visual navigation system. Theoretical analysis of registration problems is described in Sec. 3. Road structure constraints and parameterized road shape models are also provided in this section to derive a simplified linear equation of camera pose estimation. Section 4 gives the system implementation details and demonstrates the efficiency and effectiveness of proposed algorithms by the experimental results of real road tests under different conditions.

2. Review of VICNAS

Since Pioneer introduced the world's first commercial GPS car navigation system in 1990, on-road navigation related research has become one of the most active areas of many new techniques. An on-road navigation system is generally defined as the integrated system that is mainly used to provide location and navigation information to help drivers drive on road. The first generation of on-road navigation systems was equipped with the basic functionalities of 2D map displaying and road positioning. With the development of voice guidance and dynamical traffic information exchange techniques, recent navigation systems will guide you with voice instructions well in advance of your next move along a planned route. However even with the voice guidance and 2D road map, the driver still has to judge by himself the road scene with the digital map to determine which lane he should take, or, at which intersection he should turn. It is inconvenient and even dangerous in some cases, especially during high-speed driving.

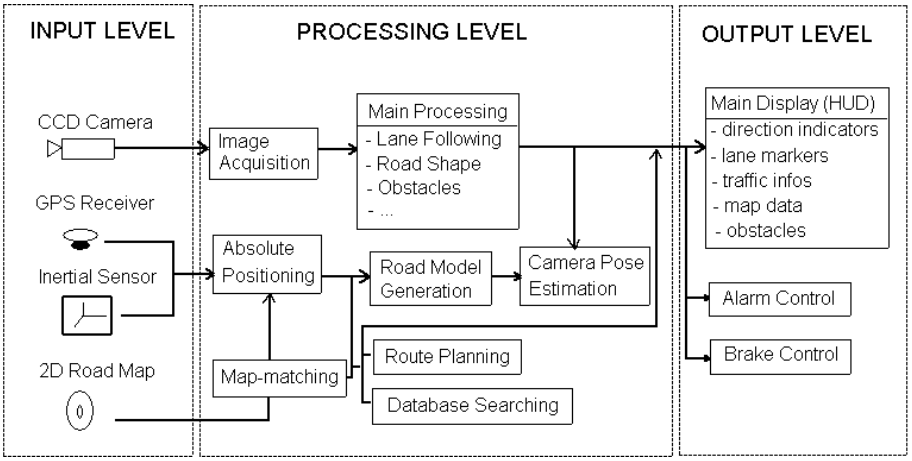


Fig. 1. System architecture of VICNAS.

To overcome these indirect, unsafe and inconvenient navigation problems, the authors proposed a new concept of direct visual navigation and its prototype system — Vision-based Car Navigation System (VICNAS).[3] VICNAS employs an image processing technique to match the digital map data with the image sequence obtained from an on-board CCD camera, and superimposes the virtual direction indicators and the traffic information bulletins into the real road scene to give drivers an efficient and direct visual navigation guidance. Figure 1 shows the basic architecture of VICNAS.

VICNAS consists of three major levels: Input level, Processing level and Output level. Image data, GPS data and road information are fed into different processing units respectively. Image data will be synchronized with GPS data and acts as the main processing source to perform lane following, road shape recognition, obstacle detection and intersection recognition. Corrected by the inertial sensor, GPS data provides the absolute positioning information to locate the car and generate road models. The generated road models will be used in the camera pose estimation step, which is the main topic of this paper. The output level consists of Head-Up Displaying (HUD), light and voice alarm control. Break, steering and throttle controls will be implemented in the future autonomous driving phase. Virtual guidance indicators, virtual road paintings, and traffic information bulletins will be superimposed upon the real road scene to provide the driver with efficient, direct and visual guidance.

3. Camera Pose Estimation

3.1. *Reference frames*

As described in Sec. 1, the general concept of vision-based camera 3D pose estimation is to find the best set of camera position and orientation data (the

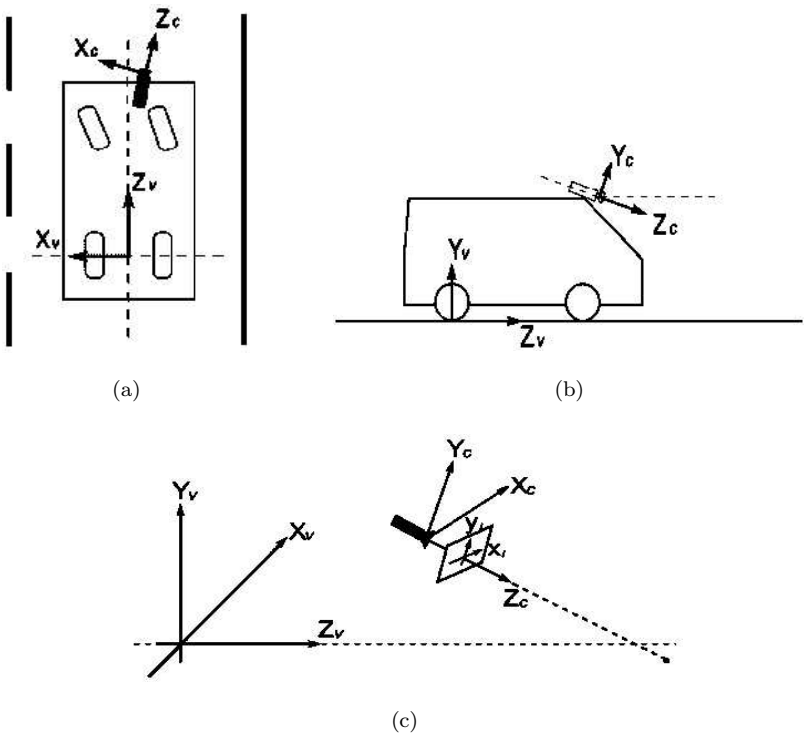


Fig. 2. The three reference frames of on-road navigation.

six extrinsic parameters) to fit a known model in the target image. In the application of on-road navigation, the process starts with the coordinates conversion between different reference frames: from the World Coordinate System (WCS) to the Vehicle Coordinate System (VCS), and then through the Camera Coordinate System (CCS) to the final 2D projected Image Coordinate System (ICS).

As shown in Fig. 2, we assume that the CCD camera of on-road navigation system is mounted on the roof of a front-wheel drive vehicle. The origin of VCS (X_v,Y_v,Z_v) is located at the center of two rear wheels on the ground plane, Z_v, which is on the vehicle's central axis, X_v, Y_v, that are pointing left and up respectively. Since WCS's origin is also located on the ground plane, VCS can also be treated as the relative WCS with an offset and heading angle on the ground (X_v,Z_v) plane.

The CCS (X_c,Y_c,Z_c) is also called the viewpoint reference frame. It is a right-hand orientated system and Z_c coincides with the camera optical. The perspective Image Coordinate System ICS (x_i,y_i) is located on the projection plane $Z_c = f$, where f is camera's focal length. Figure 2(c) shows the relative placement of three reference frames.

Mapping from CCS to ICS is a perspective projection. We will denote in this pape $\vec{P_c}$= $[X_c\ Y_c\ Z_c\ 1]^T$ as the homogeneous coordinates of a point P_c in CCS space. The corresponding point in ICS is $\vec{p_i}$= $[x_i\ y_i\ 1]^T$. A 3×4 matrix Γ represents

the perspective projection:

$$\kappa \vec{p_i} = \Gamma \vec{P_c} \tag{1}$$

where κ is an arbitrary scale factor. Γ can be decomposed into:

$$\Gamma = K \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} = K[E|0]\,, \tag{2}$$

where matrix K is called the camera intrinsic parameters matrix, and normally it can be extracted by scale factors (S_x, S_y), principle point (u_0, v_0) and screw factor S_θ:

$$K = \begin{bmatrix} S_x & S_\theta & u_0 \\ 0 & S_y & v_0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{3}$$

The transformation from CCS to VCS can be easily described as a rigid body translation and rotation. Equation (4) shows the homogeneous transformation matrix.

$$\vec{P_c} = M\vec{P_v} = \begin{bmatrix} R_{11} & R_{12} & R_{13} & T_X \\ R_{21} & R_{22} & R_{23} & T_Y \\ R_{31} & R_{32} & R_{33} & T_Z \\ 0 & 0 & 0 & 1 \end{bmatrix} \vec{P_v} = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \vec{P_v}\,, \tag{4}$$

where R and T are the matrixes of rotation and translation respectively, and matrix M is also called camera extrinsic parameters matrix. It represents the transformation from camera (viewpoint) reference frame onto the model reference frame.

A linear constraint is then held by combining Eqs. (1) to (4).

$$\kappa \vec{p_i} = \Gamma \vec{P_c} = K[E|0]M\vec{P_v} = K[E|0] \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \vec{P_v}\,. \tag{5}$$

Fixed camera intrinsic parameters can be easily obtained from the initial calibration. Therefore computing the six extrinsic parameters in the pose data matrix M becomes the major problem of camera pose estimation. If we collect the six camera pose parameters in one vector σ, we can simply parameterize the perspective mapping relationship between the 2D image coordinates in ICS and the 3D world coordinates in VCS as follows:

$$\vec{p_i} = \Gamma(\vec{P_v}; \sigma) \tag{6}$$

In the general solution of the registration problem, each matching pairs of $(\vec{p_i}, \vec{P_v})$ will contribute to the determination of camera pose vector σ. However, in the application of on-road navigation, there is no pre-defined model in the view and previous research in this literature showed that it is extremely difficult to obtain

the depth information with the image data alone. To avoid direct matching between ICS and VCS, a parameterized multi-lane road modeling method and 3D-2D road model matching algorithm are proposed this paper.

3.2. *Absolute positioning and road modeling*

VICNAS employs a hybrid-measuring method to obtain the absolute positioning data. In an open, well-communicated environment (number of tracked satellites $N_s > 5$ and value of PDOP[a] < 4.0), GPS data with differential correction information (DGPS) can have the accuracy of 1.5m horizontally and 5m in altitude. In the city area, GPS accuracy will be extremely affected by the obstruction and random reflection of signals (so-called multi-path) by the roadside trees, high-buildings and other artificial constructs. Hybrid-measuring methods, normally with inertial sensors like acceleration sensor and wheel rotation meters, are carried out to compensate the GPS data. A map matching technology is then used to pull the absolute positioning data to the nearest possible road according to the moving trace history.

The generic road information embedded in the 2D digital map consists of a list of road skeleton node positions (longitude and latitude values), road segments (between two adjacent road skeleton nodes), intersections information and the associated attributes to the road segments (road construction level, direction information and lanes number in either direction). Lane width is generally fixed and can be determined by the information of road level. Figure 3 shows an example of road skeleton, node and segment.

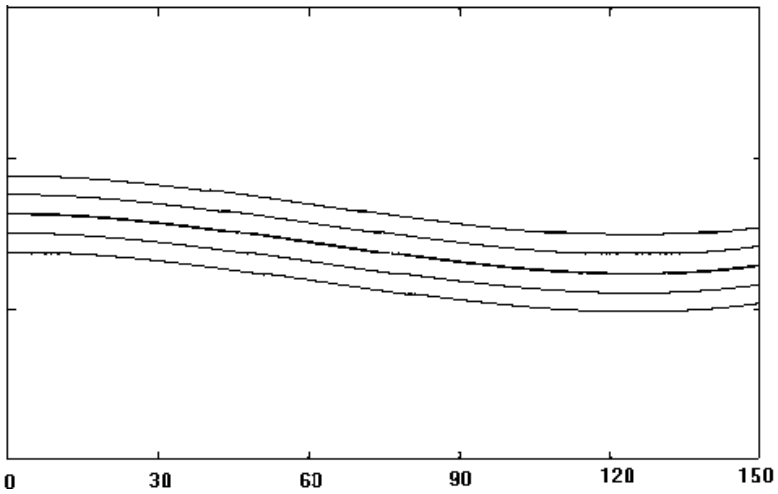


Fig. 3. Road segment example derived from clothoid multi-lane road model.

[a]PDOP (Position Dilution of Precision) indicates the factor of geometrical configuration of tracked satellites, and smaller PDOP value gives better accuracy.

The civil engineering regulations require the connections between road nodes to be smooth and the changing of curvature to follow the road level and speed limitation. A widely used road model is a clothoid whose curvature c is proportional to its length l:

$$c(l) = c_0 + c_1 l\,, \tag{7}$$

where c_0 is the curvature at the beginning and c_1 is the rate of change (derivation) for the curvature.[9] Actual roads are — as environmental conditions allow — a set of successive clothoids (including straight lines and circles). In a multi-lane case, each lane should keep the same width and maintain equal distance with neighboring lanes along the course. Road centerlines are generally two solid lines with a certain distance (except the case when central separators in the two-way road is wider than the lane width, which is always considered as two separate roads in the digital map).

This geometric description of horizontal road shape results in a very compact parameterized multi-lane road shape model on WCS of road segment i (from node i to node $i+1$):

$$\Re_i = (c_{0i}, c_{1i}, n_{li}, n_{ri}, w_i, L_i)^T\,, \tag{8}$$

where n_{li} is the number of lanes on the node-descent side (from node $i+1$ to node i), n_{ri} is the number of lanes on the node-ascent side (from node i to node $i+1$) and w_i is the average lane width during this segment. Appendix shows the detail derivation of multi-lane road model expression. For any road segments, the neighboring road skeleton node location are used to calculate the clothoid parameters c_0 and c_1.

An example of road segment is shown in Fig. 3, where $L_i = 150$ m, $n_{li} = n_{ri} = 2$, $w_i = 3.75$ m, $c_{0i} = 4.5 \times 10^{-3}$ m^{-1}, $c_{1l} = -7.2 \times 10^{-5}$ m^{-2}.

In order to obtain the road shape ahead at any driving location on the road, we have to transfer the road model $\Re_i$ from WCS at time i to a new model $\Re'_i$ in VCS according to the offset and heading angle, since the road model's origin is based on the road central skeleton line. Assuming that the road surface in the nearby view is flat and the ground plane is at $Y_v = 0$, together with the perspective mapping Eq. (6), the following perspective road model is obtained.

$$\vec{p}(\Re'_i) = \Gamma(\Re'_i; \sigma)\,. \tag{9}$$

Figure 4 shows the perspective mapping of road model in Fig. 3.

3.3. *3D-2D road model matching and camera pose tracking*

As mentioned in Sec. 3.1, it is extremely difficult to obtain the depth information from image sequence taken by a single camera. To avoid directly matching between the 3D road model $\Re'_i$ and 2D image data, we set the problem as the optimization of the matching between 2D projected road model as shown in Eq. (8) and the road shape data extracted from the image. The idea is that a perspective road model at

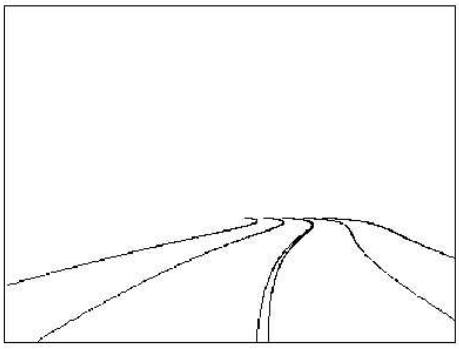


Fig. 4. Perspective projection of multi-lane road model.

certain camera pose will give the best matching result to the current image data, and we want to maximize the amount of matching pixels of perspective road model.

The matching cannot go directly into the gray level correlation due to various type and colors of road lane markers, different lighting and weather conditions as well. To counteract this effect, we present in this paper a concept of Road Shape Look-up table (RSL), which gives peak values at the position of lane marker and the values decrease at their neighbors. Therefore as a unified road shape description, RSL values will be used in the road model matching instead of gray values.

The calculation of RSL is based on the extracting result of road lane markers.[10] Suppose point $p(x, y)$ belongs to the region of extracted lane markers, its RSL value is calculated through the convolution of neighbor pixels with a predefined kernel χ:

$$RSL :\rightarrow p(x,y) = \sum_i \sum_j \lambda_{x-i,y-j}\chi_{i,j}\,, \tag{10}$$

where $\lambda_{x,y} = \begin{cases} 1, & p(x,y) \in \text{lane marker} \\ 0, & \text{otherwise} \end{cases}$.

Kernel $\chi_{i,j}$ is always used to realize some special functionalities like image smoothing, noise removing, or edges enhancing in a certain direction. The simplest smooth kernel is to set all elements as 1. The kernel size is generally set to 5 or 7 in order to have a larger base of attempts for the optimum.

Therefore, a normalized camera pose estimation score function can be given as:

$$E(\sigma) = \frac{1}{|\eta_\sigma|} \sum_{p\in\eta_\sigma} \parallel RSL :\rightarrow p(x,y) \parallel = \frac{1}{|\eta_\sigma|} \sum_{p\in\eta_\sigma} \parallel RSL :\rightarrow \Gamma(\Re';\sigma) \parallel, \tag{11}$$

where η_σ represents the set of points belonging to the revised perspective road model $\Re'$. Every point with a non-zero RSL value will contribute to the score function. In other word, the maximum of estimation score will be reached at the perfect matching of projective road model to the road shapes on the image.

In the final optimization operation, we adopt the direct search algorithm presented by Hooke and Jeeves.[11] Continuously pose tracking from the image sequence is the problem of recording and feeding back the still image estimation result over

time. At each discrete time i, the estimation result of the last frame $i-1$ will be used to update the current position and orientation together with the new GPS data, and it will also be used to provide a searching start point in the optimization.

4. Results

4.1. *Initial camera calibration*

An initial calibration test was carried out in order to calculate the camera's intrinsic parameters as well as the position relationship between the two coordinate systems VCS and ICS.

Illustration of the calibration test is shown in Fig. 5. We used 4×6 pre-defined grids on the ground as references. Vehicle rear wheels were aligned to the center of the last row which gives the VCS coordinates of each grid points as:

$$(X_v, Y_v, Z_v)^T = ((1.5 - i)d_{\text{column}}\,, \quad 0\,, \quad (5.0 - j)d_{\text{row}})^T \tag{12}$$

where row number $i = 0 \sim 3$, column number $j = 0 \sim 5$, d_{row} and d_{column} are the interval distances of row and column respectively. The ICS coordinates of each grid can be interactively or automatically obtained from the image.

We employed the Haralick's Modified Weights Method[6] to solve this typical absolute orientation problem. In order to simplify the computing, we assume that the affection of lens distortion is neglectable and the camera roll angle can be easily adjusted to 0 during the system setup. Image principle point is usually located in the center of image, which gives $u_0 = v_0 = 0$. The skew factor S_θ is also assumed to be 0. The calibration test result is shown in Table 1.

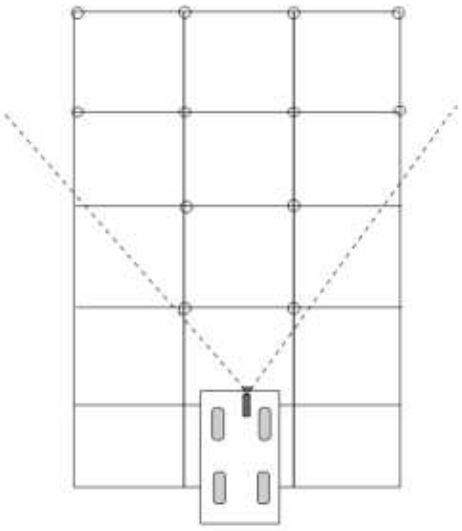


Fig. 5. Camera calibration.

Table 1. Calibration results of intrinsic and extrinsic parameters.

Notation	Description	Estimated values
f	Camera focal distance	467 pixels
T_X	Displacement on X_v axis	−12.24 cm
T_Y	Displacement on Y_v axis	197.02 cm
T_Z	Displacement on Z_v axis	213.22 cm
Ω_X	Displacement on pitch angle	0.262 rad
Ω_Y	Displacement on yaw angle	0.136 rad

Fig. 6. Camera and GPS device.

4.2. *Road test results*

As shown in Fig. 6, a 1/4-inch finger-sized CCD analog video camera was mounted on the front of test platform minivan. Image sequences were recorded in NTSC format at the frame rate of 30fps. Differential GPS data (Trimble AgGPS) and inertial data (DataTech GU-3023) were sent to PC's serial port and recorded at the frequency of 16.7Hz. The Zenrin Z-Map (Kumamoto region) was used as the 2D road map. As the Phase 1 of the VICNAS project, our real road tests were based on the off-line processing.

Road tests were carried out on different kinds of road (express toll-way, city highway, downtown street and countryside road), different lane structures (one-way or two-way, $1 \sim 6$ lanes, with or without central separators) and shapes (straight, curve, S-curve). Some road scene snapshots and generated perspective road models are shown in Fig. 7.

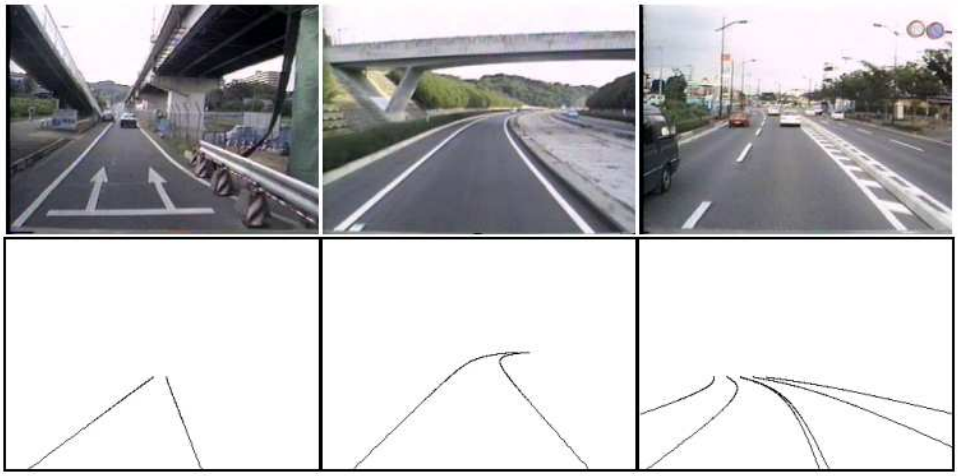


Fig. 7. Tested roads and road shape models.

flame #1 flame #39

flame #128 flame #198

Fig. 8. Tested image sequence.

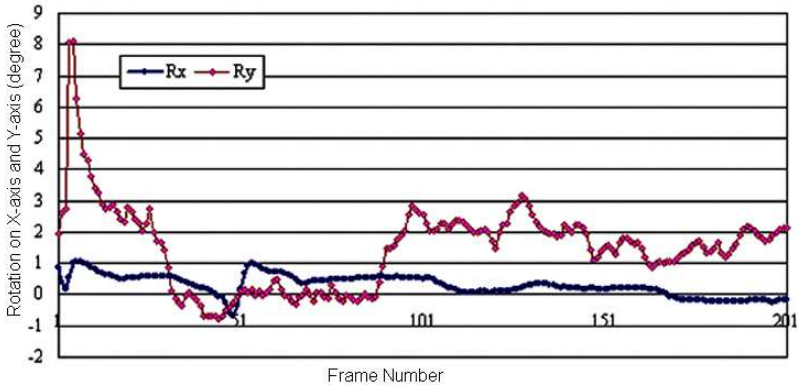


Fig. 9. Estimation results of pitch and yaw angle.

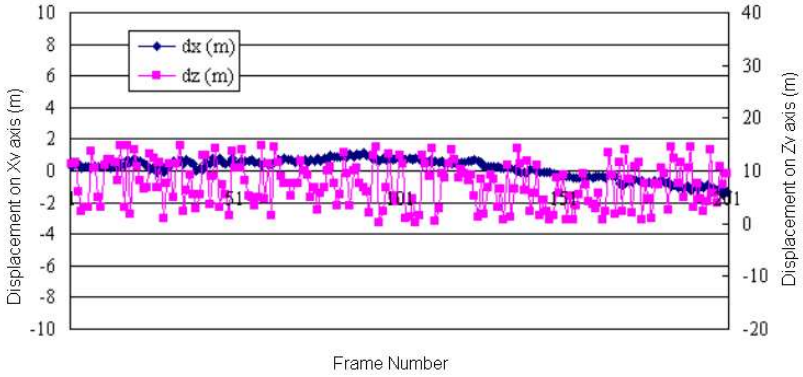


Fig. 10. Estimation results of displacement on X and Z.

As mentioned before, the camera roll angle can be easily adjusted to 0 during the system setup, and hence we will not discuss it here. Also, since the camera displacement on the Y_v axis and the camera pitch angle are always compensating each other during the estimating, we will only calculate the pitch angle and keep the displacement on Y_v as a constant.

Figures 9 and 10 show the pose estimation results from a 202 frames image sequence (Fig. 8). The GPS trace on the 2D map of this road segment is shown in Fig. 11. It is a fairly new-paved two-way road with two lanes on each direction. The vehicle was running on the central lane. Figure 9 shows the estimated camera pose angles of R_X and R_Y in the unit of degree. The results here have already eliminated the original displacement between the camera and the vehicle reference frame. A very stable R_X with an average of 0.4 degree corresponds to this flat road surface without any quick acceleration or deceleration during the drive. In the yaw angle direction, the continuous positive values of R_Y show vehicle's heading angle was left, which can be also verified by the road segments' left curve.

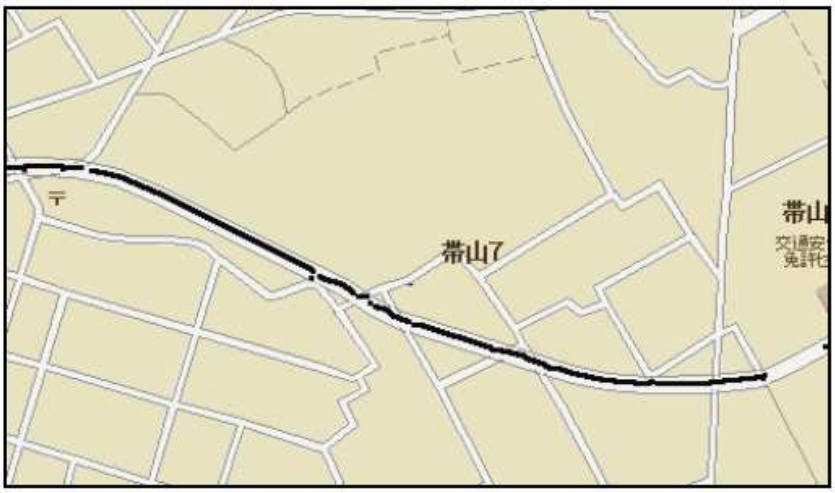


Fig. 11. GPS data trace on 2D road map.

Figure 10 shows the estimated camera position offsets on different axes. Affection of original displacement of camera to vehicle has also been eliminated. d_X gives out the distance to the road central skeleton line, which is exactly corresponding to the vehicle's position on the lane. The stable estimation result of d_X corresponds to the road scene, and it also shows good accuracy if it is compared with the GPS trace in Fig. 11. d_Z is the most unstable parameters among the estimation result. The reason is that in the geometrical feature-less road segments like the pure straight or circle road, the road shape models are exactly the same wherever the viewpoint locates. In other word, the absolute position d_Z cannot be estimated from the road model matching algorithm. We employed the absolute GPS position data to correct it.

To verify the accuracy of our estimation algorithm, a virtual direction indicator (dark gray arrow) and a virtual road bulletin (speed limitation painting) were generated and they are projected to the real road scene according to the estimated camera pose data. As shown in Fig. 12, a very smooth and stable synthesized result in the AR space verified the effectiveness of our solution to the registration problem for on-road navigation.

5. Conclusion

This paper presents a dynamical solution of the registration problem for on-road navigation applications via the 3D-2D parameterized model matching algorithm. The traditional camera's three dimensional (3D) position and pose estimation algorithms have always employed the fixed and known-structure models as well as the depth information to obtain the 3D-2D correlations, which is however unavailable for on-road navigation applications since there are no fixed models in the general road scene. With the constraints of road structure and on-road navigation

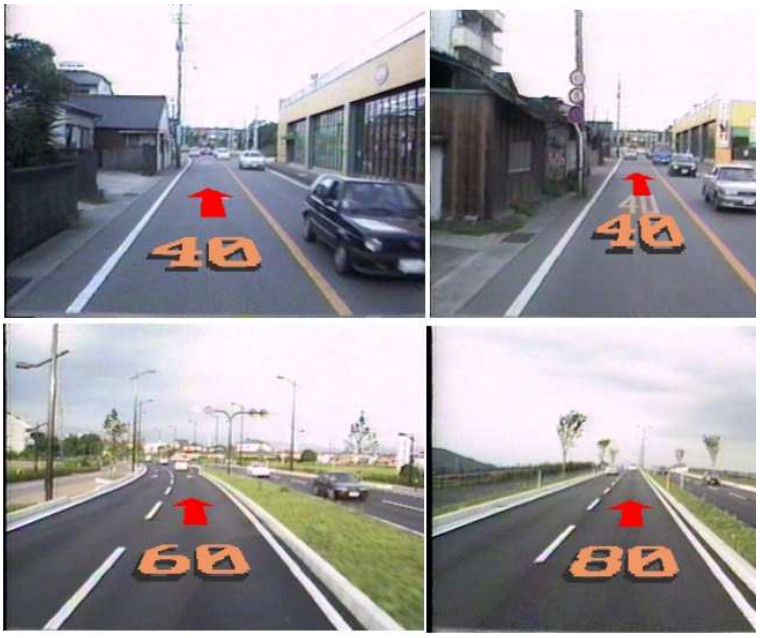


Fig. 12. Virtual indicators in the AR space.

features, this paper presents a 2D digital road map based road shape modeling algorithm. Dynamically the generated multi-lane road shape models are used to match the real road scene to estimate the camera 3D position and pose data. Our algorithms have successfully simplified the 3D-2D correlation problem to the 2D-2D road model matching on the projective image. A detailed system structure and implementation methods of the new generation of on-road navigation system (VICNAS) are also presented in this paper. Experiments on image sequences of real road scenes showed the effectiveness and the precision of our approach.

There are still some special road shape segments that are not covered by our algorithm of road model matching, such as intersection and diversion, which are also essential for on-road navigation. 3D road shape is also another interesting topic and it will become more commercially valuable when the 3D digital map data is available in the near future. Our interests will be continuously focused on these topics as well as the real-time computation and implementations in AR world.

Appendix A. Multilane Road Shape Model

As described in Sec. 3.2, the road shape consists of a set of successive clothoids, which correspond to the road skeleton nodes in the 2D digital road map. To derive a multi-lane road shape model from the clothoid shape of road skeleton line, we assume that each lane will keep the same width and maintain equal distance with the neighboring lanes along the course.

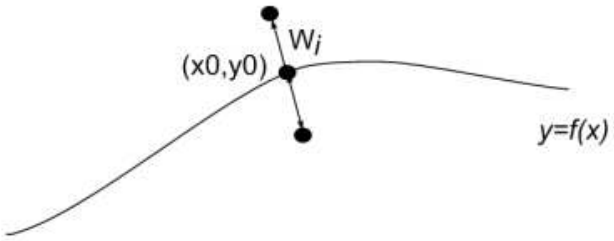


Fig. A.1. Road skeleton and multi-lane model.

Let $y = f(x)$ represents the road skeleton shape (Fig. A.1) and (x_0, y_0) is a sample point locating on the skeleton. To obtain the corresponding points on neighbor lanes on both sides, we may first take the derivation value of $f(x)$ on the sample point, which is:

$$\dot{f}(x)|_{x=x_0} = \dot{f}(x_0)\,. \tag{A.1}$$

Therefore the normal of road skeleton will be:

$$y = \psi(x) = -\dot{f}(x_0)^{-1}x + (y_0 + \dot{f}(x_0)^{-1}x_0)\,. \tag{A.2}$$

Together with the lane width constraint

$$(x - x_0)^2 + (y - y_0)^2 = w_i^2\,, \tag{A.3}$$

we can obtain the corresponding points position on the neighbor lanes:

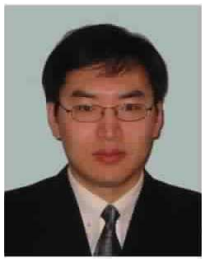
$$x = x_0 \pm \frac{w_i}{\sqrt{1 + \dot{f}(x_0)^{-2}}}, y = y_0 \pm \frac{\dot{f}(x_0)w_i}{\sqrt{1 + \dot{f}(x_0)^{-2}}}\,. \tag{A.4}$$

By changing the lane distance w_i, we will obtain the whole road shape model.

References

1. R. T. Azuma, "A survey of augmented reality," in *Presence*: *Teleoperators and Virtual Environments* **6**(4), 355–385 (1997).
2. M. Bajura, H. Fuchs and Ohbuchi, "Merging virtual objects with the real world: Seeing ultrasound imagery within the patient," *Computer Graphics*, pp. 203–210 (July 1992).
3. Z. Hu and K. Uchimura, "Tracking cycle: A new concept for simultaneously tracking of multiple moving objects," In *IEEE Intelligent Vehicles Symposium*, Detroit (MI), USA, pp. 233–239 (October 2000).
4. B. Horn and E. Weldon, "Direct methods for recovering motion," *International Journal of Computer Vision* **2**, 51–76 (1988).
5. X. Zhuang, R. Haralick and T. S. Huang, "Two-view motion analysis: A unified algorithm," *Opt. Soc. Am.* **3**(9), 1492–1500 (1986).
6. R. M. Haralick, H. Joo, R. C. Lee, X. Zhuang, V. G. Vaidya and M. B. Kim, "Pose estimation from corresponding point data," *IEEE Transactions on Systems, Man and Cybernetics* **19**(6), 1426–1446 (November–December 1989).
7. D. G. Lowe, "Fitting parameterized three-dimensional models to images," *IEEE Trans. Pattern Analysis and Machine Intelligence* **13**(5), 441–450 (May 1991).

8. A. Valinetti, A. Fusiello and V. Murino, "Model tracking for video-based virtual reality," *Proc. 11th Int. Conf. Image Analysis and Processing* (2001), pp. 372–377.
9. R. Gregor, M. Lutzeler, M. Pellkofer, K.-H. Siedersberger and E. D. Dickmanns, "EMS-Vision: A perceptual system for autonomous vehicles," *IEEE Trans. Intelligent Transportation Systems* **3**(1), 48–59 (March 2002).
10. D. Koller, T. Luong and J. Malik, "Binocular stereopsis and lane marker flow for vehicle navigation: Lateral and longitudinal control," *Report of UCB/CSD*, 94–804 (March 1994).
11. R. Hooke and T. Jeeves, "Direct search solution of numerical and statistical problems," *Journal of the Association for Computing Machinery* (*ACM*) (1961), pp. 212–229.
12. M. Bertozzi and A. Broggi, "GOLD: A parallel real-time stereo vision system for generic obstacle and lane detection," *IEEE Trans. Image Processing* **7**(1), 62–81 (January 1998).
13. A. Broggi, M. Bertozzi, A. Fascioli, C. G. Bianco and A. Piazzi, "Visual perception of obstacles and vehicles for platooning," *IEEE Trans. Intelligent Transportation Systems* **1**(3), 164–176 (September 2000).
14. A. Scheuer and Th. Fraichard, "Continuous-curvature path planning for car-like vehicles," *IEEE-RSJ Int. Conf. Intelligent Robots and Systems* (September 1997).
15. G. L. Foresti, V. Murino and C. Regazzoni, "Vehicle recognition and tracking from road image sequences," *IEEE Trans. Vehicular Technology* **48**(1), 301–317 (January 1999).
16. K. Nickels and S. Hutchinson, "Model-based tracking of complex articulated objects," *IEEE Trans. Robotics and Automation* **17**(1), 28–36 (February 2001).
17. M. Harville, A. Rahimi, T. Darrell, G. Gordon and J. Woodfill, "3D pose tracking with linear depth and brightness constraints," in *Proc. CVPR 99*, Corfu, Greece (1999), pp. 206–213.
18. Z. Hu and K. Uchimura, "Motion detection from a moving observer using pure feature matching", *International Journal of Robotics and Automation* **15**(1), 21–26 (2000).



Zhencheng Hu received his B.Eng. degree from the Shanghai Jiao Tong University, China in 1992. He received his M.Eng. degree in 1998 and his PhD degree in System Science in 2001, both from the Kumamoto University, Japan. Dr. Hu has held various positions in the computer science and machine vision industry. He is currently a senior machine vision application specialist in Matrox Electronic Systems Ltd., Canada. His research interests include camera smotion analysis, augmented reality, machine vision applications in industry and ITS. Dr. Hu is a member of the IEEE, and the Institute of Electronics and Information Communication Engineers of Japan (IEICE).

Keiichi Uchimura received the B.Eng. and M.Eng. degrees from the Kumamoto University, Kumamoto, Japan, in 1975 and 1977, respectively, and the PhD degree from the Tohoku University, Miyagi, Japan, in 1987. He is currently a Professor with the Department of Computer Science, Kumarnoto University. He is engaged in research on intelligent transportation systems, and computer vision. From 1992 to 1993, he was a Visiting Researcher at the McMaster University, Hamilton, ON, Canada. Dr. Uchimura is a Member of the Institute of Electronics and Information Communication Engineers of Japan.