

PANORAMIC REPRESENTATION OF SCENES FOR ROUTE UNDERSTANDING

Jiang Yu Zheng and Saburo Tsuji

Department of Control Engineering, Osaka University
Toyonaka, Osaka 560, Japan

Abstract

This work tackles route understanding in robot navigation. The strategies employed are route description from experience and route recognition by visual information. In description phase, a new representation of scenes along a route termed *Panoramic Representation* is proposed. It is obtained by scanning scenes sideways along the route, which provides rich information such as a 2D projection of scenes called *Panoramic View*, a path-oriented 2(1/2)D sketch, and a path description. The continuous *Panoramic View* is more efficient in processing than integrating discrete views into a complete route model. In recognition phase, the robot matches the *Panoramic Representation* from incoming images with that memorized in the previous scan so that it can locate and orient itself in autonomous navigation. The advantage of wide field of *Panoramic Views* brings reliable scene recognition.

1. Introduction

Much research on vision-based navigation has been focused on road-following and obstacle-avoidance. Their aim is to move a robot safely within a free space based on sensor data^[1,2,3,4]. For longer distance navigation, however, robots will also be confronted with another problem: how to understand the route it travels, which includes issues of sensing environment, creating spatial memory, locating robot, and selecting routes in the global world. The work described here discusses how to represent environment that a robot moves around, and how to recognize the scenes so as to locate and orient itself.

Two kinds of outdoor environment representations, a bird's-eye view map and a series of route views, can be considered^[5]. The robot can determine its way from either of them. Although a bird's-eye view map such as an aerial photograph or a city map is good for representing global relationships of routes and is straightforward in route planning, it is hard to be interpreted into real route views for recognition. A robot using a terrain map to understand its environment has been proposed^[6,7]. Another approach is to use route views. Landmarks specified by humans are employed in guiding the navigation^[8].

We propose a paradigm of describing route from experience. A sequence of ground-based views are acquired through traversing routes and is further analyzed for guidance in robot navigation^[9]. The scenario is as follows: A robot moves along a certain route under the guidance of a human and visually memorizes the scenes. It is then commanded to pursue the same route autonomously. The robot keeps observing the scene and locates and orients itself by referring to the memorized route description so that it can instruct the road-following module where it should change its direction or stop. Fig.1 illustrates a part of such a route.

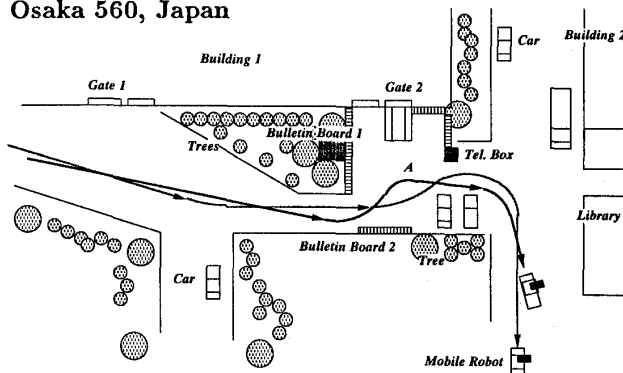


Fig.1 A mobile robot continuously views scenes along a route by a camera. It autonomously builds the model along the route, which guides the navigation along the same route. A route shown in this figure is used for the experiments.

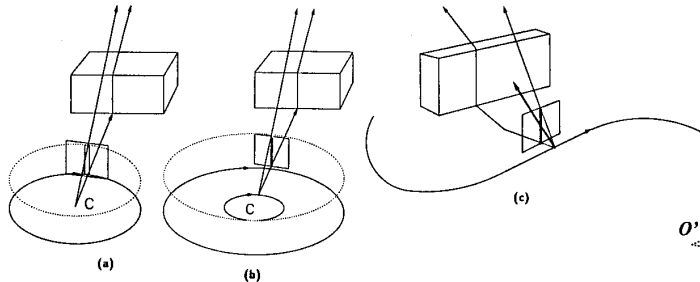
The key step to bridge local episodic scenes along a route to a global semantic description for route understanding is, perhaps, encoding of numerous route views into an easy-to-access presentation. In this paper, a novel representation of scenes, called *Panoramic Representation*, is proposed, for memorizing and retrieving the visual information acquired in the trial move. This representation provides the essential information of scenes, such as 2D projection, a path-oriented 2(1/2)D sketch, and a path description. It can be used not only as an intermediate representation as Marr proposed^[10], from which a more abstract and symbolic representation of the route can be built, but also as a description referred directly to in route recognition.

In order to represent a wide field of view that contains global information with a small amount of data, we use *Dynamic Projection*^[11,12] which scans scenes through a moving vertical slit. We call it *Panoramic View (PV)* which has the following two types:

- *Local Panoramic View (LPV)*: Projection of scenes around a stationary view point, memory of a place used for determining location and orientation of the robot.
- *Route Panoramic View* or simply *Panoramic View*: Projection of side-views along a route, memory of the route used for locating the robot.

The 2(1/2)D sketch described by the horizontal image velocity of each feature, is extracted by measuring the time delay of features in appearing in a pair of parallel slits in the image.

In route recognition phase, the robot recalls the route memorized in its trial move to identify its location and orientation. We, therefore, need to match two panoramic representations obtained from different moves along the same route, assuming the road-following process keeps the path within the roads. We also match two local panoramic views at positions close



Figs.2 Schemes for generating Panoramic Views. (a) Acquiring LPV by taking slit image by a swiveling camera. (b) A modified PV is yielded when a camera moving along a circle. (c) A generalized PV is acquired by a camera moving along a smooth path, with its optical axis aligned with the normal of path.

each other to find a correct way or to move for the exact destination.

Utilizing the sequential and continuous characteristics of the panoramic representation, we employ Dynamic Programming^[13] and Circular Dynamic Programming (a modified Dynamic Programming for periodical scenes) in matching color projections of PVs and LPVs, to find coarse correspondence. Then the distinct features appearing in 2D projections are verified in detail by examining their attributes. 2D shape changes due to changes in path are normalized by using 2(1/2)D information acquired in establishing the representation.

Because of the wide sight of PV, the matching can start from a fairly global level to avoid failure from changed parts and, thus, results in a high reliability. Color-based recognition of outdoor scenes suffers from instability by the changes in illumination. We improve constancy of colors of panoramic representations from different moves.

2. Panoramic Representation

2.1 Formation of Panoramic View

A method to model a wide environment is to direct a camera towards various parts of the environment and take discrete images. However, the spatial relationship between these images is not easy to obtain, except when they share certain common fields of view. The drawbacks of this consideration are the data redundancy and the two-dimensional discontinuity in the overlapped parts which arises from the differences of view points and viewing directions.

We take numerous narrow views at finely divided angles of orientation at a single view point for LPV, and at densely distributed positions to a given direction for PV. Figs.2 illustrate the schemes of image formation. Suppose a camera rotates at point C at a constant angular velocity on a horizontal plane and takes pictures through a vertical slit as Fig.2(a) shows. By pasting the slit views consecutively at the angle of observation, we get an LPV which contains all objects visible from point C . The velocity of each image point passing across the slit is the same because the rotation center coincides with the camera focus. The LPV is considered as a minimum 2D data set which contains all the information acquired while the camera swivels.

Let us modify this slightly so that the camera moves along a circle C at a constant speed with its direction perpendicular to the motion direction as in Fig.2(b). If we sample a vertical line at the center of each image and arrange them successively into an image, we get almost the same image as the case shown in Fig.2(a), except the object's image velocity passing across the vertical line depends on its range from the path.

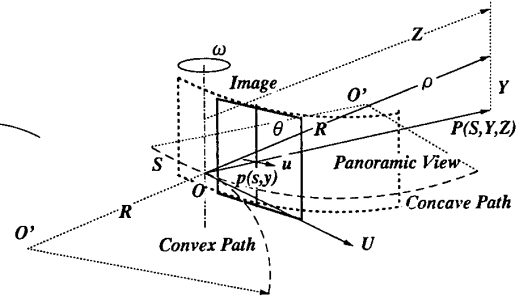


Fig.3 Camera system in generating PVs.

The case that the camera moves along a straight line can be also considered as the case where the center of the path is at infinity.

More generally, we can move the camera along any smooth curve S on a horizontal plane to obtain a panoramic view as Fig.2(c) shows. It employs a central projection along the Y axis (slit direction) and an orthogonal projection along the S axis (distance along the path).

Let us call a path segment with the centers of curvature at the visible side, at infinity, and at the invisible side as concave, linear, and convex, respectively. The constraints used for panoramic views are (1) the camera moves along a smooth curve on a horizontal plane and (2) the camera axis is horizontal and aligned with normals of the curve.

2.2 2D Projection of Panoramic View

Let us first explore properties of the projection of 3D space to the panoramic views for different camera movements. Linear and circular paths are considered, because we can approximate a smooth path by their segments. Our analyses assume the camera motions are ideal and their parameters are known. The following is notations used (see Fig.3).

- O : Camera focus.
- f : Focal length of the camera.
- U : Linear velocity of camera.
- ω : Angular velocity of camera.
- R : Radius of circular path ($R = U/\omega$).
- O' : Center of camera path (at infinity for linear path).
- S : Length passed ($S = 0$ for LPV).
- θ : Angle between the camera axis and a reference direction ($S = R\theta$).
- $P(S, Y, Z)$: A 3D point viewed at S , where Z is the depth from O , and Y is the height from the horizontal plane.
- $p(s, y)$: Projection of P in the panoramic view, where s is the coordinate of the horizontal axis.
- u, v : Horizontal and vertical components of image velocity.
- ρ : Horizontal distance of point P from the center O' (For a convex path, $\rho = R + Z$, For a concave path, $\rho = R - Z$ if P is near than O' and $\rho = Z - R$ if P is farther than O' .)
- $L(H, D)$: A horizontal line in 3D space, where D is the horizontal distance from O' , and H is the height of the line. It is also denoted by a vector $(A, 0, C)$ from an initial point (S_0, H, Z_0) for a linear path. Table 1 summarizes the 2D shapes of some basic 3D features in the panoramic views from various paths. When a horizontal line $L(H, D)$ is observed from a circular path or appears in LPV, the reference direction is selected as orthogonal to the line. The derivation of Table 1 can be found in reference^[12,14].

2.3 Path-Oriented 2(1/2)D Sketch

Motion		3D Features		Point $P(S, Y, Z)$ $\rightarrow p(s, y)$	Line		Plane	Volume	Horizontal Image Velocity of Point P	Depth of Point P
		Position			Horizontal Line $L(H, D) \rightarrow l(s, y)$	Vertical Line				
LPV $U=0$	Pure Rotation			$s=\theta, y=\frac{Y}{Z}$	$-\frac{\pi}{2} < \theta < \frac{\pi}{2}$ $y=\frac{H\cos\theta}{D}$	Vertical Line			$u=-\omega$	
	Convex Path $Z=\rho-R$			$s=(R+f)\theta,$ $y=\frac{Y}{\rho-R}$	$y=\frac{H\cos\theta}{D-R\cos\theta}$				$u=-\frac{\rho}{\rho-R}\omega=-\frac{\rho U}{RZ}$	$Z=-\frac{RU}{uR+U}=-\frac{U}{u+\omega}$
PV $U>0$	Linear Path $\omega=0, R=\infty$			$s=S, y=\frac{Y}{Z}$	$y=\frac{H}{AZ+C(s-S_0)}$				$u=-\frac{U}{Z}$	$Z=-\frac{U}{u}$
	Concave Path	$Z < R$ $Z=R-\rho$			$s=(R-f)\theta,$ $y=\frac{Y}{R-\rho}$		$y=\frac{H\cos\theta}{D+R\cos\theta}$			$u=-\frac{\rho}{R-\rho}\omega=-\frac{\rho U}{RZ}$
		$Z > R$ $Z=R+\rho$			$s=(R-f)(\theta+\pi),$ $y=\frac{Y}{R+\rho}$	$y=\frac{H\cos\theta}{D-R\cos\theta}$			$u=\frac{\rho}{R+\rho}\omega=\frac{\rho U}{RZ}$	$u > 0$ $Z=\frac{RU}{uR-U}=\frac{U}{u-\omega}$

Table.1 2D and 2(1/2)D characteristics of panoramic views from different motions.

From panoramic views, we acquire a path-oriented 2(1/2)D sketch for understanding the depth of objects along the routes, by measuring image velocities of features passing through the vertical slit.

(1) Image Velocities Observed from Different Paths

Let us examine the image velocities through a slit for camera motion along different paths. Let the focal length $f = 1$. The horizontal image velocity u of point P at the center line ($x = 0$) of the image caused by camera motion is shown in Table 1. Each object velocity in the image is inversely proportional to the range of the object for linear camera motion. The image velocity increases by a factor of ρ/R if the path is convex. For a concave path, the velocity decreases by the same factor. The points nearer than the center O' move in the direction reverse to the camera motion, while the points more distant than the center move in the same direction as the camera.

If the robot motion type and parameters are known and the horizontal image velocity of a point at the center line is measured, its depth can also be computed as in Table 1.

(2) Acquiring Image Velocities

Let us assume a stable robot motion, such that both motion class and velocity are invariant, at least for a short period. The horizontal image velocity of a feature through the image center line is estimated as its average velocity between two vertical slits placed symmetrically to the center line. Generating two PVs from the same images through the two slits, we determine the velocity from the difference of the horizontal positions of the feature in two PVs.

Figs.4 intuitively show the traces of a point P in the images for different paths. Because of the symmetry of the path to the line of sight, the point moves on a line or circles relative to camera, which is line or ellipses symmetric to the central line in images. Then, the feature appears at the same height on the two slits. A constraint for matching becomes

C1: A point P has the same y coordinates in two PVs generated from a pair of parallel slits symmetric to the central vertical line.

2.4 Route Description

The robot speed U , angular velocity ω , and the sampling rate τ are recorded with PV. They influence on the panoramic representation as the scale changes but not on the route geometry. The intrinsic parameters of the route geometry are:

- Distance S along the route from the start point or selected landmarks.
 - Curvature of the path described by radius of curvature R .
- These can be estimated from U and ω as:

$$S = \int U dt, \quad R = U/\omega \quad (1)$$

The length t of a panoramic view is determined by route length S and an image sampling time τ . For constant U and τ , we simply have

$$S = U\tau t \quad (2)$$

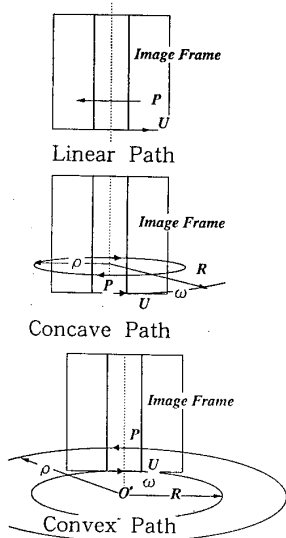
3 Acquiring Stable Panoramic Representations

3.1 Acquiring Panoramic Representation from the Real World

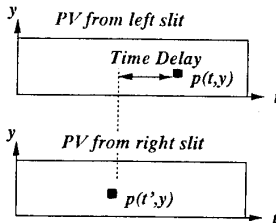
Experiments were made in the route shown in Fig.1 using a mobile robot (an automatic guided vehicle for factory automation) with a color TV camera, which can be swiveled on a rotating table. An on-board microcomputer, which controls the robot and camera motion, communicates with a Sun4 workstation with an image processor.

Fig.5 displays an LPV of the outdoor environment taken at a corner on the route, in which one can find movable areas. Fig.6(a) shows an example of a Panoramic View of size 2048 x 128 pixels acquired while the robot moved about 100 meter. The sampling rate was constant and the route consisted of an almost linear path followed by a concave and then convex paths. The robot stopped midway for a little while, waiting for passage of an obstacle, which yields an area covered with horizontal stripes in the PV. Unevenness of the road causes variations in the camera pitch, which results in much zigzagging in horizontal lines.

By setting two vertical slits $L1, L2$ at $(-\Delta x'/2, \Delta x'/2)$ symmetric to the center line in the frame, we generate two PVs.



(a) Traces of a point in the field of view when PVs from linear, concave and convex paths are generated.



(b) Acquiring image velocity by matching the PVs from the same images. Fig.4 Constraint for matching two PVs from slits symmetric to the center line.

Only vertical lines are analyzed because they will not break due to the unevenness of road while their heights are influenced. Fig.6(b) shows the matched vertical lines in the PVs. The matching method will be described in Chapter 4. By finding the duration Δt between matched pairs in the PVs, we obtain the time of lines penetrating $L1, L2$, which are displayed by the length of horizontal segments attached on them. The horizontal image velocity u at the center is computed as $\Delta x' / \Delta t$.

We do not intend to use the robot speed U and ω to estimate the position of arrival because of the error accumulation. Our attempt is to build a more flexible route model in which approximate geometry is described. The robot locates itself by referring to the scenes on the route, assuming it can move along an almost identical path. The followings are defined for instructing a road-following process to pursue the memorized route.

- (1) Break point, where the curvature of path exceeds a certain threshold and is considered as a corner to turn, is attached to the PV. A qualitative direction such as [Leftward, Rightward] is assigned. The entire route is thus divided into sub-routes connected at these break points.
- (2) Complex break point, where an open area or more than three ways are ahead, to which a local panoramic view is attached for determination of direction to move.

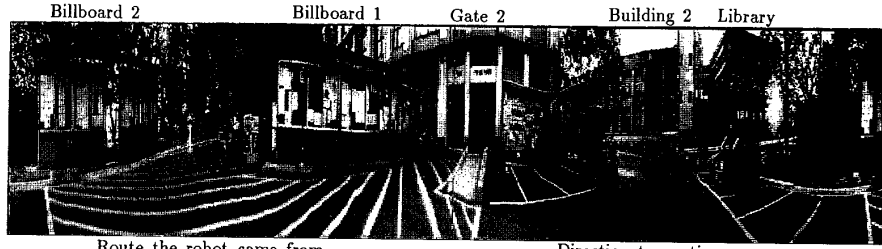


Fig.5 LPV taken by a camera on a robot turning at point A on the route shown in Fig.1.

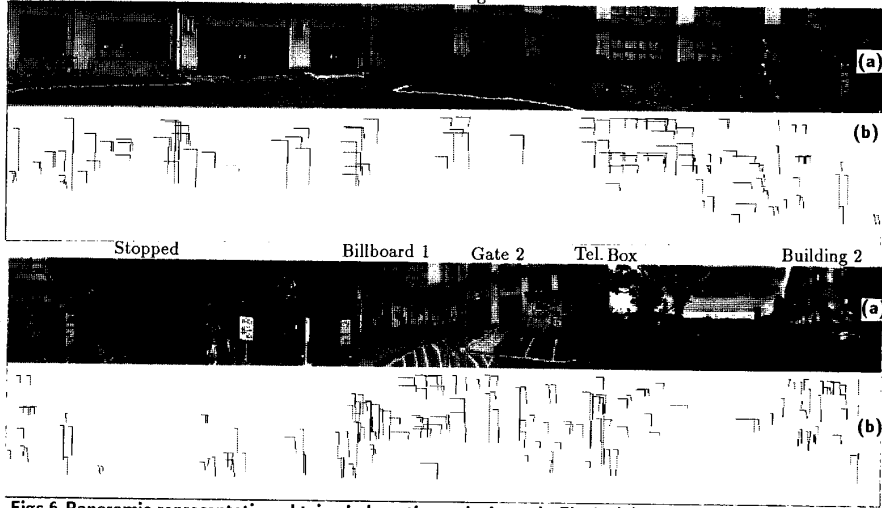


Fig.6 Panoramic representation obtained along the path shown in Fig.1. (a) Panoramic view. (b) Vertical lines in the PV and their depth information. The horizontal line segments attached to them indicate their time delays in the PVs from two vertical slits.

3.2 Color Constancy Improvement

In the real outdoor environment, there is drastic change in the illumination. The panoramic representation therefore should be robust to the change. In order to understand object color from sensor data, *Color Constancy* has been studied [16,17,18]. In this work, we apply the retinex theory [17], which thus far has been applied only in the ideal laboratory scene, to the outdoor panoramic views. Rather than being concerned with the recovering the absolute material reflectance, we explore an efficient method for improving color constancy to some extent so as to allow the reliable matching of scenes in route recognition.

Intensive studies have explained that the spectral radiance sensed at an object point is the product of the surface reflectance and the spectral distribution of incident light. The method hence is to generate a virtual white by averaging spectral distribution in a large field of view. The chromaticity of illuminant is bearing in the spectrum of the field. Then, we remove the constant illumination component by normalizing the color spectrum of each pixel by the virtual white in the entire view. This method requires many distinct color surfaces visible in the image, but it is not so hard to achieve in the outdoor environment, because we can enlarge the region for processing to a wide field in PV.

4 Matching Two Panoramic Representations

4.1 Coarse Matching using Dynamic Programming

In real navigation, it is difficult for a robot to pursue the

exact same path in different moves. The panoramic representations are yielded in a path-oriented fashion. Fortunately, we can assume that the paths will not be separated too far apart so that we can match the two representations. Otherwise, the robot will conclude that it has gone on a different route. Analysis of shape changes in PVs due to the changes in paths has been given^[14].

The matching process is coarse-to-fine, it first matches the color projections $h(t)$ and $h'(t')$ of two PVs onto the t axes, and, then, precisely matches vertical lines $l(i), i = 1, \dots, n$ and $l'(j), j = 1, \dots, m$ in both PVs, by searching narrow regions around the locations determined by the color matching. Because patterns in PV change in the t -scale due to the changes in robot velocity and sampling rate of camera, Dynamic Programming (DP) methods can cope with the coarse matching of two PVs.

The color projection $h(t) = [R(t), G(t), B(t)]$ of a PV represents the average color of the vertical lines.

$$R(t) = \sum_{y=0}^{128} r(t, y), \quad G(t) = \sum_{y=0}^{128} g(t, y), \quad B(t) = \sum_{y=0}^{128} b(t, y) \quad (3)$$

The color of a large pattern in PV is the same dominant color in $h(t)$. Also, a long vertical line $l(i)$ appears as a strong "edge" in $h(t)$. After $h(t)$ is computed, we smooth it so as to maintain the most remarkable color changes in it. The edges $E(i)$ and $E'(j)$ in the projections are used as elements of correspondents, and the color values between them are used in computing an evaluation function.

Fig.7 shows a matching result of the color projections of two PVs generated from different moves. The positions where a pair of edges are matched are connected by lines. Fig.8 shows the search paths in the matching space of Dynamic Programming, in which the grid is drawn at the positions of edges $E(i), E'(j)$. The local evaluation function $p(n-1, n)$ from node $n-1$ to node n in the matching space is selected as:

$$p(n-1, n) = \sum_{t_{n-1}}^{t_n} \|h(t) - h'(t')\| + \|E(i_n) - E'(j_n)\| \quad (4),$$

where

$$\|h(t) - h'(t')\| = |R(t) - R'(t')| + |G(t) - G'(t')| + |B(t) - B'(t')| \quad (5)$$

and $\|E(i_n) - E'(j_n)\|$ is the difference of color edge strengths of two edges $E(i_n), E'(j_n)$. t' is the linear transform from t in the small part (t_{n-1}, t_n) as

$$t' = (t - t_{n-1})(t'_{j_n} - t'_{j_{n-1}})/(t_{i_n} - t_{i_{n-1}}) \quad (6).$$

4.2 Feature Matching by Attributes

Given two sequences of data, Dynamic Programming can find the optimal correspondence between them. The question of whether the matching result of color projections fits the real scenes in the panoramic views has to be checked further. After the approximate positions of patterns are obtained by DP, vertical lines near the positions are easily matched by comparing their attributes such as edge strength, colors of both sides, etc.^[7].

Since the viewing positions in different moves are different, 2D shapes in two PVs may be inconsistent. However, we can normalize feature candidates to those viewed at the same depth by using the acquired 2(1/2)D information, in order to check the consistency in 2D size. Two additional constraints available for line matching in PVs are as follows:

Let Y_1, Y_2 denote the Y coordinates of terminals of an almost vertical line in 3D space. Their y coordinates in a panoramic view are y_1, y_2 and the distance from the camera is Z . We have

$$y_1/Y_1 = f/Z, \quad y_2/Y_2 = f/Z \quad (7a)$$

Thus,

$$y_1/y_2 = Y_1/Y_2 \quad (7b)$$

Since Y_1/Y_2 is a constant while the camera axis is horizontal, we have

C2: The ratio of the heights of two end points y_1/y_2 for an almost vertical line is invariant in PVs while its depth changes.

Let $L = Y_1 - Y_2$ denote the length of a line observed from distances Z and Z' in different moves, and its projected lengths in two PVs be l and l' . The ratio l/l' gives another constraint.

C3: For a vertical line, the lengths of its projections in two PVs have the relation:

$$\frac{l}{l'} = \frac{Z'}{Z} \quad (8)$$

where Z'/Z is computed from formulae in Table 1 using motion parameters and the image velocity u .

We examine the lengths of candidates in addition to the other properties. Thus, 2(1/2)D information is used in adjusting the size changes of shape in matching the scenes viewed at different points. Two lines $l(i), l'(j)$ in different PVs are accepted as a matched pair if the most similar candidate of $l(i)$ is $l'(j)$, and vice versa. Fig.9 displays the matched pairs of line.

4.3 Circular Dynamic Programming

Matching of two local panoramic views obtained by swiveling the camera at two locations is slightly different from matching PVs. First, any initial position in the LPVs where the matching can start is not given since the robot direction at that location is uncertain, and we have to consider all the possible combinations of features in both LPVs. Second, a stable result can be achieved by iterative matching, because of the circular structure of the LPVs. We thus introduce a method termed *Circular Dynamic Programming* algorithm applicable to matching general circular distributions. After the color projections of the LPVs onto the direction axis are matched, vertical lines in them are checked to verify the correspondence.

Fig.10 depicts the basic idea of circular dynamic programming. Suppose two edge sets $E = [E(i), i = 1, \dots, n]$ and $E' = [E'(j), j = 1, \dots, m]$ in two color projections are obtained from the corresponding LPVs. The optimal matching should establish a closed path of one period in the search space of DP.

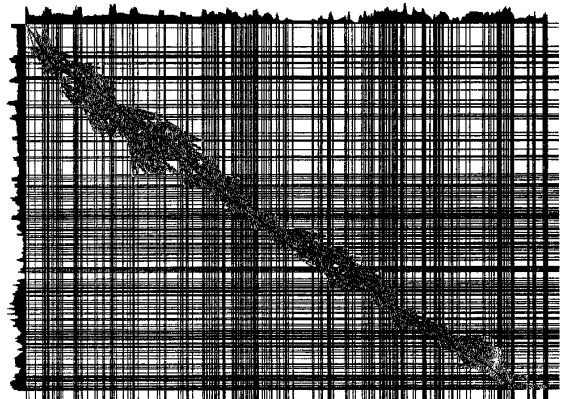


Fig.8. Searching for the optimal path in Dynamic Programming space. $R(t)$ and $R'(t')$ of the projections are displayed at the left and top margins. The grid is drawn at edges positions. Gray paths indicate the possible matching and the dark one indicates the optimal correspondence.

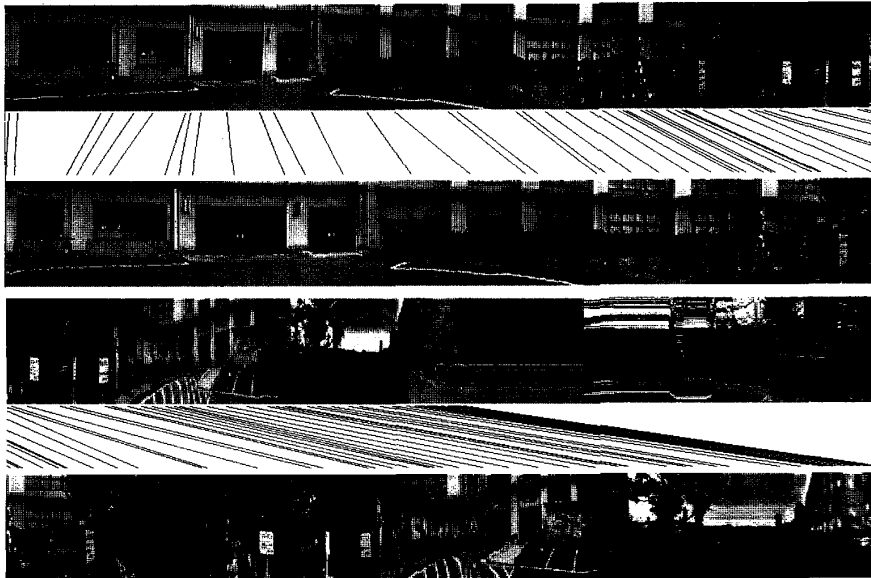


Fig.7. Matching color projections provides a coarse correspondence of two PVs from different moves.

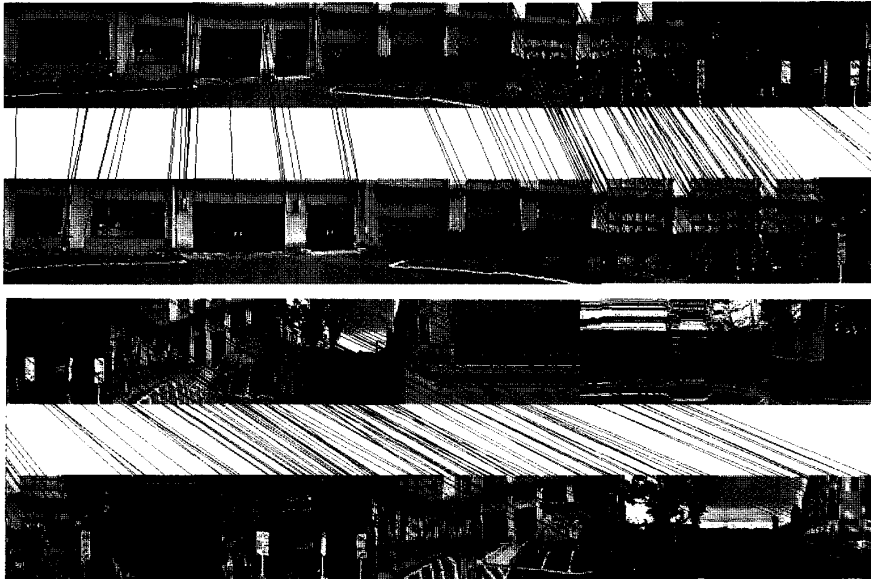


Fig.9 Matching of vertical line segments in two PVs from different moves.

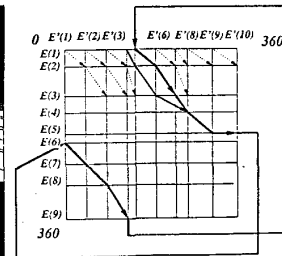
The circular dynamic programming works as follows.

- (1) The search starts with all the possible combinations $[E(1), E'(j)]$, $j = 1, \dots, m$.
- (2) In order to save memory and computation, only a selected number of nodes is expanded, and the expanded path is dynamically substituted for a sustained path if its cumulative evaluation value is inferior (known as beam search).
- (3) After a path arrives at the end $E'(m)$ of the edge set E' , it continues to search from the beginning edge $E'(1)$ so that a circular search is realized.
- (4) When a search path arrives at the end $E'(n)$ of the edge set E' , it extends search to the starting edge $E(1)$, which brings a reliable matching selection back to the beginning where

the combinations taken into consideration in step (1) may be uncertain.

- (5) The search paths expand iteratively across the matching space until the optimal path by that step forms a closed curve over one period.

The circular iterative matching modifies the optimal path to pass through more correct corresponding positions, which yields a stable result on the coarse level. Fig.11 displays 4 LPVs taken at positions separated by 1.5m, 6m, and 10m from each other to the right of point A in Fig.1 and their matching results. Figs.12 shows the search paths of possible matching and the obtained closed curves representing the optimal matchings. Matched results are also shown as lines



Searched Path: (1,4)-(2,5)-(3,6) - (4,8)-(5,9)-(5,10)-(6,1)-(7,2)-(8,3)-(9,4)-(1,5)-(2,6)-(3,7)-(4,8)

OptimalPath:(4,8)-(5,9)-(5,10)-(6,1)-(7,2)-(8,3)-(9,4)-(1,5)-(2,6)-(3,7)-(4,8)

Fig.10 Circular Dynamic Programming searching for the optimal closed path of corresponding.

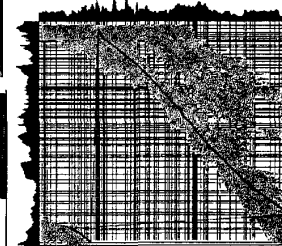
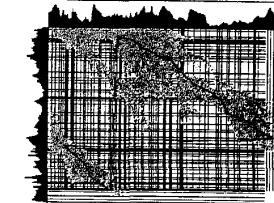


Fig.12 Matching spaces of the LPVs (LPV3 to LPV4, LPV4 to LPV1) in which closed curves represent the optimal correspondences.

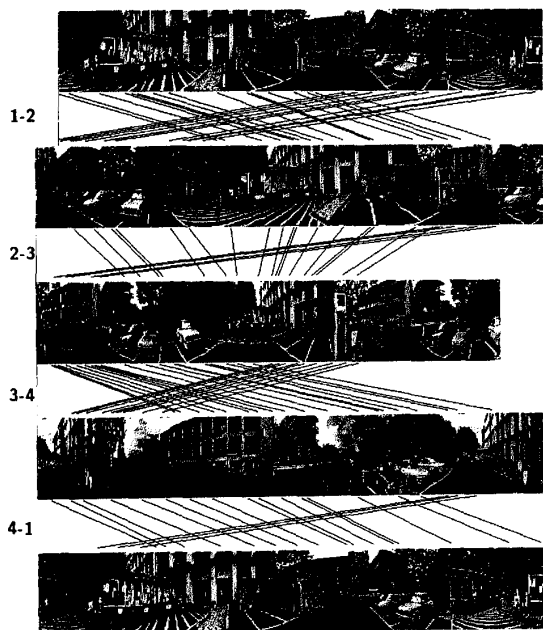


Fig.11 Matching LPVs at four positions by Circular Dynamic Programming. Matching results 1-2, 2-3, 3-4, and 4-1 are depicted by lines connecting them.

drawn between the LPVs. If two positions imaging LPVs are distant from each other, objects in LPVs may suffer from a severe size change and occlusion. Failure in matching of such LPVs means that the robot is needed to approach that position.

5 Route Pursuit

There are two ways to guide the road following process in route pursuit. As the robot moves along a street in an urban area, the destination is possible to be occluded by objects intervening and the heading of robot is approximately parallel to the road. The direction to move is able to be described simply as *Ahead*, *Turn left*, *Turn right*. The route recognition task only needs to notice specific positions in panoramic view to stop or turn. On the other hand, if the robot enters a wider area such as a square or a complex break point, there are more movable spaces and possible headings. The robot can mark the destination on LPVs and approach it by adjusting its heading.

In the cognition phase, there is no particular demand in processing time, since the entire processing can be done off-line after scenes are recorded. In automatic navigation, however, a quick response to incoming scenes is necessary. Because of the small amount of data in a PV, realizing the real-time recognition is promising.

Since the matching of the panoramic views is implemented on iconic level, we have to deal with some inconsistency between the PVs from different scans. Dynamic objects may pass by and interfere with the static objects. Generally speaking, the probability of their appearance in the PV or LPV is much lower than in discrete images, because only one line in the image is sampled at any instance. An object having high relative speed appears in PV within several lines of pixel, and we can use the $2(1/2)D$ information to eliminate them by finding salience in image velocity with static background.

Objects such as parked cars may also appear in one PV but disappear in the other. If the size of a changed part is small, it will not disturb correct matching of long PVs on coarse level, because the DP evaluates the correspondence from the accumulated value over a long range. If a changed part is large, the matching may fail and the robot will get lost. One idea for solving this problem is that it restarts matching with the confused part, where the matching evaluation is low, by a similar method as that for LPV.

6 Conclusion

This paper presents a dynamically generated panoramic representation for route recognition by a mobile robot. The described issues are visual sensing, spatial memory, and scene recognition. The panoramic representation is established from continuous viewing through a slit. It provides a continuous 2D memory with a small amount of data, which maintains essential information about shape and location of scenes along a route. The panoramic views can also yield a $2(1/2)D$ sketch from the horizontal image velocity at the slit.

In route recognition, we studied the matching of two PVs (also LPVs) generated from slightly different paths or positions such that the robot can identify its location and orientation for guiding the road-following process. Since the PV covers a wide field of view, we can achieve a reliable matching using coarse-to-fine method, starting from a very coarse level. To solve the problems of color and shape changes due to the different illumination and path, we improve color constancy and employ the $2(1/2)D$ information in the matching.

References

- [1] A.M. Waxman, J.J. LeMoigne, B. Scinvasan, "A visual navigation system for autonomous land vehicles" IEEE J. Robotics and Automation, vol.RA-3, no.2, pp.124-141, 1987.
- [2] C. Thrope, M.H. Hebert, T. Kanade and S.A. Shafer, "Vision and navigation for the Carnegie-Mellon Navilab", IEEE Trans. PAMI, vol. PAMI-10, no.3, pp.362-373, 1988.
- [3] A. Elfes, "Sonar-based real-world mapping and navigation", IEEE J. Robotics and Automation, vo.RA-3, no.3, pp.249-265.
- [4] S. Tsuji, and J.Y. Zheng, "Visual Path Planning" in Proc. Int. Joint Conf. Artif. Intell.-87, Vol.2, pp.1127-1130, 1987.
- [5] G. Cohen, "Memory in the real world", Lawrence Erlbaum Associates, Hove, 1989.
- [6] D.T. Lawton, et al., Terrain models for an autonomous land vehicle, Proc. IEEE Conf. Robotics and Automation, pp.2043-2051, 1986.
- [7] T.S. Levitt, D.T. Lawton etc. "Qualitative Navigation" DARPR Proc. Image Understanding Workshop, pp.447-465, 1987.
- [8] B. Bhanu, H. Nasr and S. Schaffer "Guiding an autonomous land vehicle using knowledge-based landmark recognition" DARPR Proc. Image Understanding Workshop, pp.432-439, 1987.
- [9] J.Y. Zheng, M. Asada, and S. Tsuji, "Color-based panoramic representation of outdoor environment for a mobile robot" Proc. 9th-ICPR vol.2, pp.801-803, 1988.
- [10] D. Marr, Representing visual information, in: A. Hanson and E.M. Riseman, Eds., Computer Vision Systems, Academic Press, New York, 1978.
- [11] J.Y. Zheng, and S. Tsuji, "Spatial representation and analysis of temporal visual events", Proc. IEEE Int. Conf. ICIP-89, Vol.2, pp.775-779, 1989.
- [12] J.Y. Zheng, and S. Tsuji, "From Anorthoscope Perception to dynamic vision", Proc. IEEE Int. Conf. Robotics and Automation, May, 1990.
- [13] Y. Ohta and T. Kanade, "Stereo by two-level dynamic programming" Proc. IJCAI-85, Vol.2 pp.1120-1126, 1985.
- [14] J.Y. Zheng, "Dynamic Projection, Panoramic Representation, and Route Recognition", Ph.D. thesis in Osaka University, Dec. 1989.
- [15] H. Baker and R.C. Bolles, "Generalizing epipolar-plane image analysis on the spatiotemporal surface" Proc. CVPR-88 pp.2-9, 1988.
- [16] B.K.P. Horn "Determining lightness from an image" Computer Graphics and Image Processing (1974) 3, pp.277-299, 1974.
- [17] E.H. Land "Recent advances in retinex theory" Vision Res. 26, 7-21 (1986)
- [18] M. D'Zmura, P. Lennie, "Mechanisms of color constancy" JOSA-A Vol.3, No.10. pp.1662-1672, October 1986.