

REAL TIME ALGORITHMS FOR OBSTACLE AVOIDANCE BY USING REPROJECTION TRANSFORMATION

Zhigang Zhu and Xueyin Lin

Lab of Computer Vision and Robotics
 Department of Computer Science, Tsinghua University
 Beijing 100084, P.R.China

ABSTRACT

The picture captured by a tilted camera fixed on the top of a mobile robot can be mapped by a reprojction as if the camera is facing down. The reprojected picture has many interest properties. Based on the principle of reprojction transformation and the analysis of motion, two realtime algorithms for detecting obstacles in front of the mobile robot are developed and compared with each other. The algorithms have been implemented on PIPE Model 1 system and tested by real scene image sequences.

1. INTRODUCTION

While a mobile robot moving on a plane ground, the tasks such as path planning, obstacle avoidance and navigation, are all based on sensing and understanding its environment in real time. Usually the camera is fixed on the top of the mobile robot with a tilt angle looking down to the ground in order to gather information on its roadway. In such situation all the objects in the scene seem to be moving under the constraint that the locus of each surface point lies on a plane parallel to the ground. Based on this observation, we find that if the picture captured by the camera is mapped into a plane parallel to the ground as if the camera is fixed facing down, the mapped pictures will present many interesting properties which can be used to facilitate the work of feature correspondence and / or structure recovery needed to fulfil many vision tasks (e.g. obstacle avoidance). As the mapping likes a reprojction exerting on the pictures, it is called reprojction transformation. In the next section, the geometric principle of *reprojction transformation* is explained and some of the properties are introduced.

2. REPROJECTION TRANSFORMATION

2.1. Basic Equations and Concepts

The basic coordinate systems are robot centered coordinate system (RCC), camera centered coordinate system (CCC), and image coordinate system (uv) as shown in Fig. 1. The camera is mounted on the top of a mobile robot with a tilt angle α looking down. The origin of CCC is at the optical center of the optical system, its Z axis (optical axis) coincides with z axis of RCC, and X axis is parallel to the ground. The origin of RCC is at the bottom of the robot with its xoz plane on the ground. The location of origin of CCC with respect to RCC is (0, H, 0). Under this definition, the relationship of (X,Y,Z) in CCC and (x,y,z) in RCC for a space point can be expressed as follows:

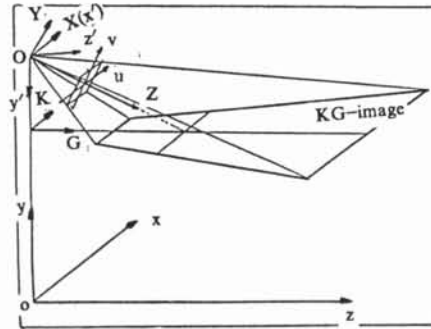


Fig. 1. Coordinate systems and geometry of reprojction transformation

$$\begin{cases} X = x \\ Y = z \sin\alpha + (y - H)\cos\alpha \\ Z = z \cos\alpha - (y - H)\sin\alpha \end{cases} \quad (1)$$

Based on geometry of pin hole model the image coordinates of a point in space can be expressed as

$$\begin{cases} u = f_u X / Z \\ v = f_v Y / Z \end{cases} \quad (2)$$

where f_u and f_v are used to account for the different scalar factors in directions u and v respectively.

From equations (1) and (2) the x and z coordinates of a point in RCC can be expressed as a function of u, v with coordinate y as follows :

$$z = G[v](H - y) \quad (3)$$

$$x = K[u,v](H - y) \quad (4)$$

where

$$G[v] = (f_u \cos\alpha + v \sin\alpha) / (f_v \sin\alpha - v \cos\alpha) \quad (5)$$

and

$$K[u,v] = (u / f_u)[G[v]\cos\alpha + \sin\alpha] \quad (6)$$

If we rearrange equations (4) and (3) as

$$\begin{cases} K[u,v] = x' / y' \\ G[v] = z' / y' \end{cases} \quad (7)$$

where $x' = x, y' = H - y$ and $z' = z$, and compare

equation (7) with (2), it is interesting to notice that if $K[u, v]$ and $G[v]$ are used as the coordinates of the pictures instead of u and v respectively, the original picture array will be mapped into a trapezoidal one (Fig. 1), which looks as if captured by a camera fixed facing

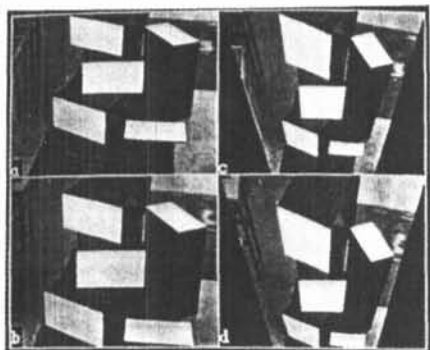


Fig. 2. Reprojected pictures of a pair of images taken while the mobile robot is moving forward. (a),(b) are first and second original images and (c),(d) are the first and second KG-images.

down with the $x'y'z'$ coordinate system. We call the mapped picture as *reprojected picture* and the mapping as *reprojection transformation*, which is expressed by equations (5) and (6). From then on as a matter of convenience, we also call the reprojected pictures as KG-images (and use K and G instead of $K[u, v]$ and $G[v]$).

2.2. Properties of Reprojection Transformation

A pair of original real images taken during the translational egomotion and their reprojected pictures are shown in Fig. 2. Some of the interesting properties of the reprojection transformation with potential usage are as follows:

. *Horizontal shape invariance*: It is intuitive from equation (7) that the geometric shapes of the figures lying on the horizontal plane remain unchanged in the KG-image except that a scalar factor related to the height of the plane is involved. Obviously the property of shape invariance can be used to structural shape detection, model-based recognition and etc..

. *KG-image invariance under Ground Plane Assumption(GPA)*: Suppose that all the figures in sight are on the ground, then the two regions in a pair of KG-images corresponding to the same region of the scene would keep unchanged while the mobile robot is doing a translational and/or rotational motion on the ground. So the decision of obstacle existence can be made based on verifying image invariance.

. *Parallel motion field constraint*: In case of translational motion, the optical flow field will appear to be a parallel field in KG-images. If structure from motion method is based on feature correspondence, the searching for corresponding feature will be reduced to one dimensional problem under parallel motion field constraint, just as epipolar constraint used in parallel stereo⁽³⁾.

. *Height from egomotion*: While only translational

egomotion of mobile robot exists, the height of each surface point can be estimated based on equation (7). In a period of time two pictures are taken, the displacements of mobile robot are expressed as T_x and T_z along x and z axis respectively, then the displacements of each feature point in a pair of KG-images can be calculated from equation (7) as

$$\Delta K = T_x / y' \quad (8)$$

$$\text{and } \Delta G = T_z / y' \quad (9)$$

From either of equation (8) and (9), the height of each point can be estimated without heavy computation. By taking advantage of this observation, a method called *Height From Motion* (HFM) has been developed by Zhu and Lin⁽⁴⁾ to gather 3D information from planar motion.

3. REALTIME OBSTACLE DETECTION

In this section two methods of obstacle avoidance using KG-images are introduced and their realtime implementations on PIPE are discussed. The first method is based on Ground Plane Assumption (GPA), and the second method is based on parallel motion field constraint and Height from Motion.

3.1. GPA-based Algorithm

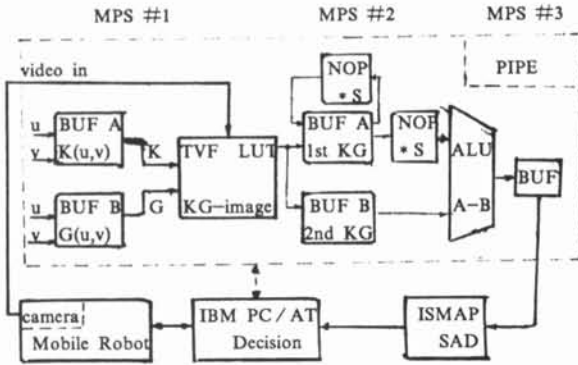
Algorithm Description: The method based on *Ground Plane Assumption(GPA)* is to verify if the shapes of all the figures in the *area of interest* (AOI) remain unchanged except for a displacement in a pair of KG-images, which are obtained during the translational egomotion. Suppose that the displacement of the mobile robot can be known precisely, then the displacement of ground figures in the KG-image pair can be calculated as

$$\Delta G = T_z / H \quad \text{and} \quad \Delta K = T_x / H \quad (10)$$

So if the first KG-image is shifted with an offset of ΔG and ΔK in G and K directions respectively, and subtracted by the second KG-image, the difference image will be with zero values everywhere theoretically in the AOI of the ground figures. Consider that the displacement of the mobile robot can not be known exactly and noise exists in KG-images. The *sum of absolute difference* of each pixel (SAD) in the AOI should be rather small if all the figures are really on the ground. Otherwise, a large residual will be expected, indicating that obstacles exist in the way.

Realtime Implementation on PIPE: Experiments have been done with a Hero-2000 mobile robot on which a 16 mm CCD camera is mounted. 256x256x8 images are processed by the PIPE, a parallel processor specifically designed for executing low level vision tasks in realtime, and driven by an IBM personal computer PC/AT. The implementing diagram of the algorithm is shown in Fig. 3 and explained as follows:

(1). The mapping from u, v to K, G are calculated off-line using equations (6) and (5), and then transfer



Notes :

- (1) * S : convolution kernel(s) used for shift the image
- (2) TVF LUT: Two Value Function Look Up Table
- (3) NOP : Neighborhood OPeration (convolution)
- (4) MPS : Modular Processing Stage
- (5) ISMAP : Iconic to Symbolic MAPper

Fig. 3 Diagram of GPA-based algorithm on PIPE

to frame buffer A and B of MPS #1, which are used as two address look up tables.

(2). A pair of images are taken during the robot's moving, and then transformed into KG-images (in TVF of MPS #1) in realtime (1 / 60 seconds), which is stored then in two image buffers of MPS #2. The first KG-image has been shifted for the amount of $(\Delta K - \delta K, \Delta G - \delta G)$ during the motion, where ΔK and ΔG are calculated from equation (10), and δK and δG are the estimated error range of ΔK and ΔG respectively.

(3). The first KG-image is shifted within a small ranges ($\pm \delta K$ and $\pm \delta G$) in K and G and subtracted by the second KG-image for each shift, using the realtime subtraction operation of PIPE. A realtime histogram operation is done in ISMAP for each subtraction to calculate the SAD in the AOI.

(4). If the minimum SAD of all the subtractions is smaller than a threshold value, it means that no obvious obstacle exists in the path way, and the more precise motion parameters could be obtained, corresponding to the minimum SAD. Otherwise the alarm is given to avoid collision.

Two examples are shown in Fig. 4 and Fig. 5 respectively. In Fig. 4, only several white marks scatter on the ground, but in Fig. 5 a stool and two bags are added. The sums of absolute differences (if more than 30) are 93529 and 2097535 in Fig. 4 and Fig. 5 respectively. The pixel numbers in the AOIs with absolute differences greater than 50 are 720 and 25022 respectively. The latter is one order larger than the former, indicating obstacle appear in the way. The current implementation requires only about 0.5 seconds for a detection cycle, including 20 subtraction operations between two KG-images.

3.2. GTI-based Algorithm

Suppose that the mobile robot is translating along z axis. In this case the motion direction of each surface

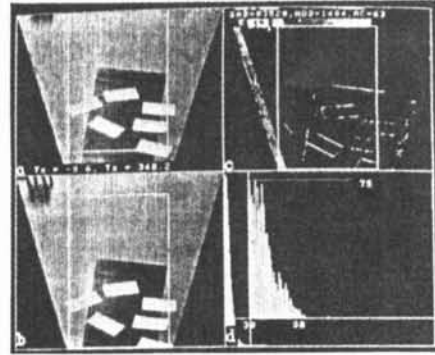


Fig. 4. GPA-based obstacle detection: example 1.

(a). shifted KG-image of frame 1 with adjusted T_x (-9.6 mm) and T_z (348.2 mm); (b). KG-image of frame 2 ;(c).subtraction of KG-images in (a) and (b); (d) the result of histogram of the AOI.

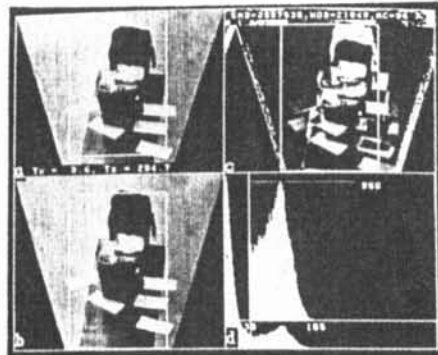


Fig. 5. GPA-based obstacle detection: example 2. Obstacles exist on the way.

point will be parallel to G axis in KG-image sequence, and the higher the point is above the ground, the larger the velocity it has in G direction. Based on this observation, the KG-image sequence can be tracked along each line parallel to G axis. For the need of real time application, only several lines parallel to G axis are taken as samples to detect obstacles. In order to get rid of the difficulty for feature correspondence, we use spatio temporal cross section image (we call it GT-image) as Bolles⁽¹⁾ and Cipolla⁽²⁾ did. The implementation on the PIPE is similar to that of GPA-based algorithm except that different LUTs and operations are used⁽⁵⁾.

Fig. 6a and 6b show the first and last KG-images of a tracking procedure, and Fig. 6c shows the three GT-images, where the horizontal direction represents time and vertical direction G value of each surface point. From GT-image, the locus of each feature point can be easily tracked and fitted (Fig. 6d), and the slope of the loci ($\Delta G / \Delta t$) can be used to estimate the relative height of each feature point. From equation (9) we have

$$\Delta G = Tz / y' = v \Delta t / (H - h) \quad (11)$$

where v is the velocity of the mobile robot in the time period Δt , and the height of each feature point can be calculated as

$$h = H - v / (\Delta G / \Delta t) \quad (12)$$

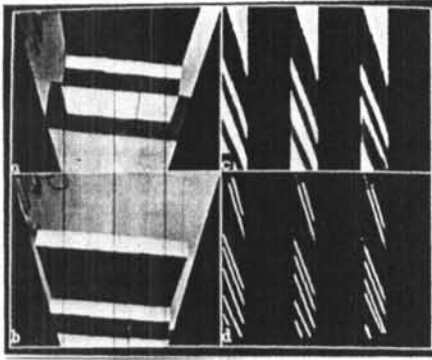


Fig. 6. GTI-based obstacle detection: (a),(b) first and last KG-images; (c) three GT-images; (d) feature loci tracking and fitting in GT-images.

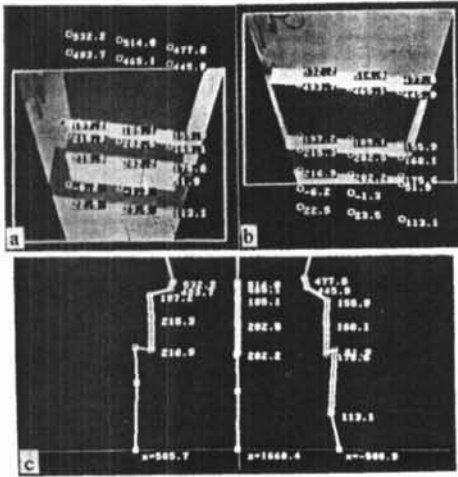


Fig. 7 Heights (mm) of feature points on the first and second KG-images. No problem is introduced with the feature disappearance (in (a)) and reappearance (in (b)).

(c). Coordinates of feature points (heights labeled on xoz map)

Furthermore the 3D coordinates of this point in RCC can be obtained by using equations (3), (4) and (12). The coordinate values of each measured feature point are shown on Fig. 7, and most of them are reasonable and accurate enough.

For convenience, we call the algorithm as GTI-based algorithm where GTI represents GT-Images. GTI-based algorithm can be easily extended to the case of translational motion which is not along z axis, i.e. $T_x \neq 0$ and $T_z \neq 0$. In this case GT-image can be constructed by tracking the KG-image sequence in the direction parallel to the direction of translation of mobile robot.

3.3. Discussion and Comparison

A brief discussion and comparison of the GPA-based algorithm and GTI-based algorithm is given here. Both of them have the following advantages: (1). Correspondence problem is avoided. (2). They can deal with any shapes of objects in any background (indoors or outdoors). (3). No problem is introduced with the presence of occlusion, disappearance and reappearance of objects features. (4). Both of them are robust and fast, and realtime performance can be

expected.

The two algorithms have their own characteristics. GPA-based algorithm has the advantage that 1) it is insensitive to the error of motion and camera calibration; 2) it operates on the whole AOIs and therefore makes use of more spatial information and 3) it can deal with motion objects (and even nonrigid objects) in the scene while the mobile robot is moving, as the SAD value would be large in this situation. But GPA-based algorithm is difficult to give the positions and heights of the obstacles, and the efficiency will be reduced with very inaccurate motion parameters. Besides, there exist some special cases where the KG-images of non-ground figures also keep unchanged⁽⁵⁾, though they are hardly encountered in practice.

GTI-based algorithm is more efficient than GPA-based one in the sense of using continuous temporal information, and it can estimate the heights and positions of all the detected feature points. But the efficiency will be reduced if more GT-images are considered. And the algorithm is more sensitive to motion error than GPA-based one.

It is obviously that the combination of the two algorithms will be better to fulfil the task of obstacle avoidance, and there is no problem to implement it on PIPE.

4. SUMMARY AND CONCLUSION

In this paper, the principle of the reprojection transformation is presented which can be used to facilitate the visual tasks of a mobile robot moving on a relative plane ground. Many important properties and the applications of this transformation are studied and discussed. Based on reprojection transformation and motion analysis, two realtime algorithms for detecting obstacles in front of the robot are developed. Correspondence problem is avoided in both of them. Robustness and high efficiency has been proved by the realtime implementations on the PIPE with real scene time varying images. The advantages of reprojection transformation will be further investigated.

Reference

- (1) Bolles, R. C., Baker, H. H. and Marimont, D.H., "Epipolar plane image analysis: an approach to determine structure," *Int. J. Comput. Vision*, Vol. 1 No. 1, p7-55, 1987.
- (2) Cipolla, R. and Masanobu, Y., "Stereoscopic tracking of bodies in motion," *Image and Vision Computing*, vol. 8, no 1, p85-90, 1990.
- (3) Lin, X.Y., Zhu, Z.G., "Detecting height from constrained motion," 3rd International Conference on Computer Vision, Osaka, Japan, Dec. 4-7, 1990.
- (4) Zhu, Z.G., Lin, X.Y., HFM: a robust and efficient method for detecting structure and constrained motion, Pacific Rim International Conf. on Artificial Intelligence, Naogoya, Japan, Nov. 14-16, 1990.
- (5) Zhu, Z.G., Lin, X. Y., "Principle and applications of Reprojection Transformation in robot vision," Technical Report, Lab of Computer Vision and Robotics, Tsinghua University, 1990.