MVA '96 IAPR Workshop on Machine Vision Applications, November. 12-14, 1996, Tokyo, Japan

3D Information Acquisition From Plane Spatio-Temporal Images with Complicated Camera Motion

Pingtao Wang * Chunxiao Li * Heitou Zen † Masao Sakauchi *
*Institute of Industrial Science, University of Tokyo, Tokyo, Japan
†Tokyo University of Mercantile Marine, Japan

Abstract

This paper presents a new method for estimating parameters of 3D lines from Plane Spatio-Temporal Image(PSTI). A PSTI is created by using a slit to scan a 3D STI which is obtained by a camera mounted on a moving car. We have discussed the creation of PSTI with the camera motion of straight-line translation by using a hyperbolical slit and the 3D information acquisition before. In this paper, we give a general analysis about the formation of PSTI at first, then propose a new method to estimate parameters of 3D lines for more complicated scanning formula and camera motion. In this method, the locus, which corresponds to a 3D line, is extracted to calculate projection vectors. And, the parameters of 3D lines are obtained from the relation between projection vectors and the parameters by recursive optimizing algorithm. The limitation of both the slit and camera motion will be decreased. At the end, some experiment results are given to reveal the effectiveness and accuracy of the method.

1 Introduction

Recovering 3D information from a 3D STI or a sequence of images is one of the main issues in computer vision. An appealing way of solving this problem is to use 2D motion computed in a 3D STI acquired by a monocular camera. Tomasi and Kanade[1] developed a factorization method to recover shape and motion under an orthographic projection model, and obtained robust and accurate results. A. Azarbayejani and P. Pentland[2] presented a formulation for recursive recovery of motion, pointwise structure, and focal length from feature correspondences tracked through an image sequence. Many other approaches have been also proposed and surveyed in Huang's review paper[5] and other survey papers.

Basically, two main approaches have been investigated to solve the problem of structure from motion: long range motion-based methods and short range motion-based ones. In the former one, images are considered at distant time instants and a large camera translation is generally performed to obtained accurate results. But inter-frame correspondences of features is difficult. In the latter approaches, images are considered at video rate. In this case, inter-frame correspondences become easy, but the data is very large and the processing become time-consuming.

Baker and Bolles developed a technique which cut EPI(Epipolar-Plane Images) along the optical flow directions in a 3D STI[3], which gives both traces of features and spatio-temporal events such as occlusion of an object by another and scene structure from motion. Li and others[6] presented another method which scan a 3D STI to create a Plane Spatio-Temporal Image(PSTI) with some transformation such as hyperbolical function. Both methods create a 2D image which includes sufficient spatial and temporal information to be used to acquire 3D information, and have not the shortcomings possessed by general structure from motion methods, but restrict the camera motion along a straight line although the former one cope with some rotation.

In computer vision, strict restriction on camera motion is undesirable. In this paper, we discuss the representation of a 3D line in a PSTI involving the camera motion at first, and present a new algorithm which can estimate parameters of 3D lines from a PSTI and have almost no limitation on both the slit and camera motion.

This paper is organized as follows. Section 2 describes the creation of a PSTI from a 3D STI. Section 3 addresses mathematical representation of a 3D line in a PSTI. Section 4 presents the method for estimating parameters of 3D lines. Section 5 provides some experimental results. Section 6 terminates the paper with some discussion.

2 The Creation of PSTI

A PSTI can be simplely thought as a section of a 3D STI. For example, both a EPI image [3] and a dynamic representation of a 3D STI [4] can be thought as special PSTIs. A PSTI which holds linear transformation for some 3D straight lines is also proposed in our former paper[6]. In fact, a PSTI

*Address: 7-22-1 Roppongi, Minato-ku, Tokyo 106, Japan. E-mail: wang,lcx,sakauchi@sak.iis.u-tokyo.ac.jp
†Address: 2-1-6 Ettyujima, Koutou-ku, Tokyo 135, Japan. E-mail: zen@ipc.tosho-u.ac.jp

is created as follows: (1) selecting a proper slit or sampling curve, (2) sampling along the slit, (3) arranging data to creating a PSTI. A slit or a sampling curve can mathematically be shown as a point set S

$$S = \{(x_{it}, y_{it}) : 0 \leq i < N, 0 \leq t < F\} \qquad (1)$$

here, N is the number of sampling points, and F is the maximum frame number. If S is related with t, that means the slit is changing with the frame number, otherwise, the slit is a static one. If a 3D STI is expressed as

$$f(x, y, t)$$

the created PSTI becomes

$$g(i, t) = f(x_{it}, y_{it}, t), \qquad (x_{it}, y_{it}) \in S \qquad (2)$$

As an example, Figure 1 shows some frames of a computer generated 3D STI, and the created PSTI with a hyperbolical slit(figure 2) is shown as figure 3.

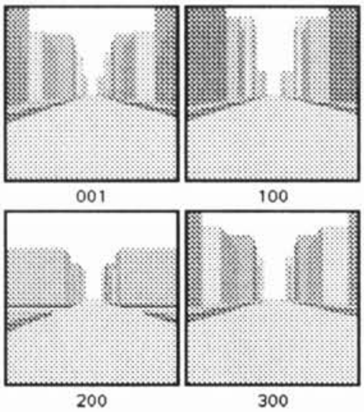

Figure 1: Frame 1, 100, 200, 300 of a 3D STI

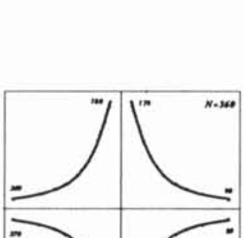

Figure 2: A composite hyperbolical slit

3 Representation of 3D Lines

A 3D straight line is mathematically expressed as

$$L : \boldsymbol{P} \times \boldsymbol{D} = \boldsymbol{P}_o \times \boldsymbol{D} \qquad (3)$$

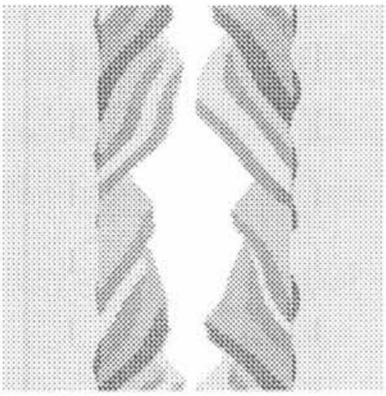

Figure 3: Plane Spatio-Temporal Image

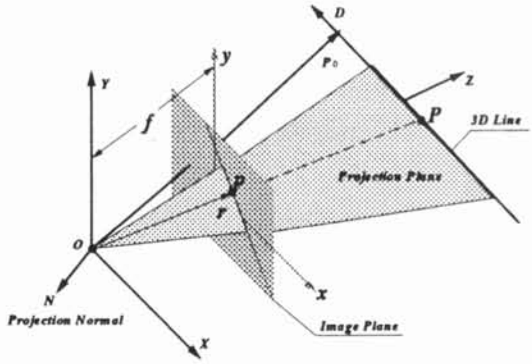

Figure 4: Projection schema of a 3D line

where, $\boldsymbol{D}$ is the direction vector of the 3D line, $\boldsymbol{P}_o$ and $\boldsymbol{P}$ are separately the vector of a fixed point and the vector of moving point on the 3D line. The projection schema is shown as figure 4. $OXYZ$ is the world coordinate system, and also the original coordinate of camera. Plane oxy is the image plane, f the focal length, and the parameters of the 3D line $\boldsymbol{Po}$ and $\boldsymbol{D}$ are also shown in the figure. Some concepts are introduced here to make following description be more simple:

Projection Vector: the vector which points from the center of camera O to the image point p of a 3D point P.

Projection Plane: a 3D line's projection plane is defined as the plane which passes both the camera's center and the 3D line.

Projection Normal Vector: a 3D line's projection normal vector is defined as the normal vector of its projection plane, and expressed mathematically as

$$\boldsymbol{N} = \boldsymbol{P}_o \times \boldsymbol{D} \qquad (4)$$

When a 3D line projects to camera's image plane, all points on the line will pass the camera's center and form a series of projection vectors such as the vector $\boldsymbol{r}$ in figure 4.

When the camera moved with rotation matrix $\boldsymbol{R}_t$ and translation vector $\boldsymbol{T}_t$ at time t relative to its original position, the projection normal vector becomes

$$\begin{aligned} \boldsymbol{N}_t &= \boldsymbol{R}_t(\boldsymbol{P}_o + \boldsymbol{T}_t) \times (\boldsymbol{R}_t \boldsymbol{D}) \\ &= \boldsymbol{R}_t \boldsymbol{N} + \boldsymbol{R}_t(\boldsymbol{T}_t \times \boldsymbol{D}) \end{aligned} \tag{5}$$

On the other hand, any projection vector of points on the 3D line $\boldsymbol{r}_t$ is perpendicular to the projection normal vector. That is

$$\boldsymbol{r}_t \cdot \boldsymbol{N}_t = 0 \tag{6}$$

We can get the relationship between the 3D line's parameters $\boldsymbol{N}$, $\boldsymbol{D}$ and the projection vectors $\boldsymbol{r}_t$ as follow

$$(\boldsymbol{R}_t{}^{-1}\boldsymbol{r}_t) \cdot \boldsymbol{N} = ((\boldsymbol{R}_t{}^{-1}\boldsymbol{r}_t) \times \boldsymbol{T}_t) \cdot \boldsymbol{D} \tag{7}$$

4 The Estimation of Parameters of 3D Lines

In general, A 3D line becomes a curve or locus in PSTI. The loci are extracted with common edge detector here. If the locus is expressed by image plane's coordinates as (x_{ti}, y_{ti}) at time t_i, the corresponding projection vectors are $\boldsymbol{r}_{ti} = (x_{ti}, y_{ti}, f)^T$, where, $0 \le i < F$ is frame No. For every projection vector $\boldsymbol{r}_t$, there is a equation which is similar to equation (7). The equations can be rewritten in a matrix form as

$$\boldsymbol{AN} = \boldsymbol{BD} \tag{8}$$

Where, both $\boldsymbol{A}$ and $\boldsymbol{B}$ are $F \times 3$ matrices:

$$\begin{aligned} \boldsymbol{A} &= \begin{pmatrix} (\boldsymbol{R}_{t0}^{-1}\boldsymbol{r}_{t0})^T \\ (\boldsymbol{R}_{t1}^{-1}\boldsymbol{r}_{t1})^T \\ \cdots \\ (\boldsymbol{R}_{tF-1}^{-1}\boldsymbol{r}_{tF-1})^T \end{pmatrix} \\ \boldsymbol{B} &= \begin{pmatrix} ((\boldsymbol{R}_{t0}^{-1}\boldsymbol{r}_{t0}) \times \boldsymbol{T}_{t0})^T \\ ((\boldsymbol{R}_{t1}^{-1}\boldsymbol{r}_{t1}) \times \boldsymbol{T}_{t1})^T \\ \cdots \\ ((\boldsymbol{R}_{tF-1}^{-1}\boldsymbol{r}_{tF-1}) \times \boldsymbol{T}_{tF-1})^T \end{pmatrix} \end{aligned} \tag{9}$$

Considering the relation between projection normal vector and direction vector

$$\begin{cases} \boldsymbol{D}^T\boldsymbol{D} = 1 \\ \boldsymbol{D}^T\boldsymbol{N} = 0 \end{cases} \tag{10}$$

an object function can be defined as follow

$$\begin{aligned} f(\boldsymbol{D},\boldsymbol{N}) &= (\boldsymbol{AN} - \boldsymbol{BD})^T(\boldsymbol{AN} - \boldsymbol{BD}) \\ &\quad + \lambda_1(\boldsymbol{D}^T\boldsymbol{N})^2 + \lambda_2(\boldsymbol{D}^T\boldsymbol{D} - 1)^2 \\ &= \boldsymbol{N}^T\boldsymbol{A}^T\boldsymbol{AN} - \boldsymbol{N}^T\boldsymbol{A}^T\boldsymbol{BD} \\ &\quad + \boldsymbol{D}^T\boldsymbol{B}^T\boldsymbol{BD} - \boldsymbol{D}^T\boldsymbol{B}^T\boldsymbol{AN} \\ &\quad + \lambda_1(\boldsymbol{D}^T\boldsymbol{N})^2 + \lambda_2(\boldsymbol{D}^T\boldsymbol{D} - 1)^2 \end{aligned} \tag{11}$$

here, λ_1 and λ_2 are coefficients. Let the values of the derivative of $f(\boldsymbol{D},\boldsymbol{N})$ with respect to both $\boldsymbol{D}$ and $\boldsymbol{N}$ be zero respectively

$$\begin{aligned} \frac{\partial f(\boldsymbol{D},\boldsymbol{N})}{\partial \boldsymbol{N}} &= 2(\boldsymbol{A}^T\boldsymbol{A})\boldsymbol{N} - 2(\boldsymbol{A}^T\boldsymbol{B})\boldsymbol{D} \\ &\quad + 2\lambda_1(\boldsymbol{D}^T\boldsymbol{D})\boldsymbol{N} \\ &\equiv 0 \\ \frac{\partial f(\boldsymbol{D},\boldsymbol{N})}{\partial \boldsymbol{D}} &= 2(\boldsymbol{B}^T\boldsymbol{B})\boldsymbol{D} - 2(\boldsymbol{B}^T\boldsymbol{A})\boldsymbol{N} \\ &\quad + 2\lambda_1(\boldsymbol{N}^T\boldsymbol{ND}) + 4\lambda_2(\boldsymbol{D}^T\boldsymbol{D} - 1)\boldsymbol{D} \\ &\equiv 0 \end{aligned}$$

then, we can get the relation between $\boldsymbol{D}$ and $\boldsymbol{N}$

$$\begin{cases} \boldsymbol{N} = (\boldsymbol{A}^T\boldsymbol{A} + \lambda_1(\boldsymbol{D}^T\boldsymbol{D}))^{-1}(\boldsymbol{A}^T\boldsymbol{B})\boldsymbol{D} \\ \boldsymbol{D} = (\boldsymbol{B}^T\boldsymbol{B} + \lambda_1(\boldsymbol{N}^T\boldsymbol{N}) + 2\lambda_2(\boldsymbol{D}^T\boldsymbol{D} - 1))^{-1} \\ \qquad (\boldsymbol{B}^T\boldsymbol{A})\boldsymbol{N} \end{cases} \tag{12}$$

The parameters $\boldsymbol{D}$ and $\boldsymbol{N}$ can be obtained by using optimal algorithm from equation (12). But, it's obvious that $\boldsymbol{D} = 0$ and $\boldsymbol{N} = 0$ is also the solution of equation (12). In order to avoid the worst case, two angles are used to substitute the three elements of $\boldsymbol{D}$ in the optimizing process. In fact, direction vector $\boldsymbol{D}$ can be expressed as the function of two angles α and β

$$\boldsymbol{D} = \begin{pmatrix} \sin\alpha\cos\beta \\ \sin\alpha\sin\beta \\ \cos\alpha \end{pmatrix} \tag{13}$$

where, α is the angle between the 3D line and Z-axis, β the angle between X-axis and the projection of the 3D line on OXY plane. And equation (12) becomes

$$\begin{cases} \boldsymbol{N} = (\boldsymbol{A}^T\boldsymbol{A} + \lambda_1(\boldsymbol{D}^T\boldsymbol{D}))^{-1}(\boldsymbol{A}^T\boldsymbol{B})\boldsymbol{D} \\ \boldsymbol{D} = (\boldsymbol{B}^T\boldsymbol{B} + \lambda_1(\boldsymbol{N}^T\boldsymbol{N}))^{-1}(\boldsymbol{B}^T\boldsymbol{A})\boldsymbol{N} \end{cases} \tag{14}$$

5 Experiments

We conducted experiments using computer generated data. Some frames of the 3D STI are shown as figure 1, and figure 3 shows the created PSTI with a composite hyperbolical slit shown in figure 2. The camera moved without rotation and along the following route

$$\boldsymbol{T}_f = \begin{pmatrix} 2.5\sin(5\pi f/256) \\ 0.5\sin(8\pi f/256) \\ 0.3 + 0.1\cos(3\pi f/256) \end{pmatrix}$$

The corresponding velocity curve are shown in figure 5.

Figure 6 shows the plan view of the generated street. The solid lines show the buildings' edges, the dashed lines show the camera's route, and the rectangle marks show the extracted buildings' corners.

Figure 7 and figure 8 show the estimating error of direction vectors and relative error of position parameters of some 3D lines respectively. The horizontal axes of both figures are line Numbers, and the vertical axes are errors.

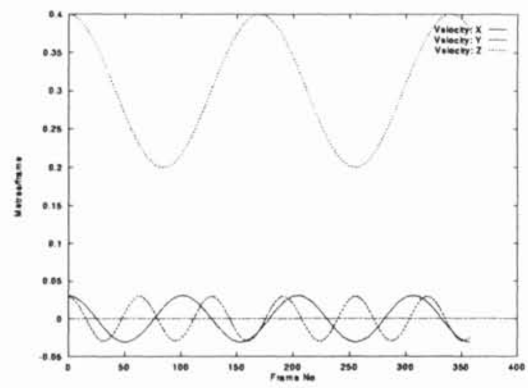

Figure 5: Camera's velocities

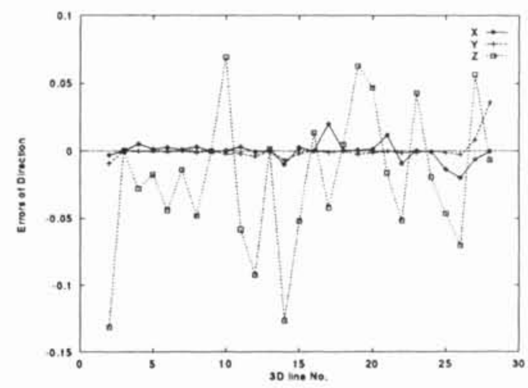

Figure 7: Estimating error of 3D line's direction

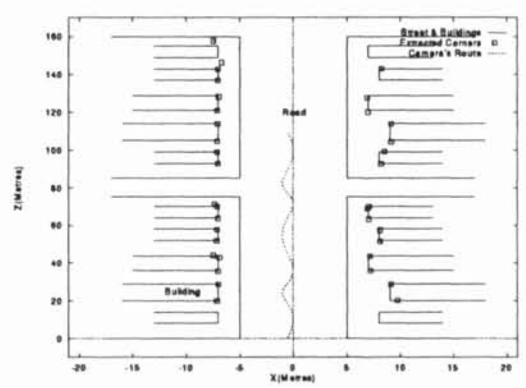

Figure 6: The plan view of the generated street with extracted positions of buildings

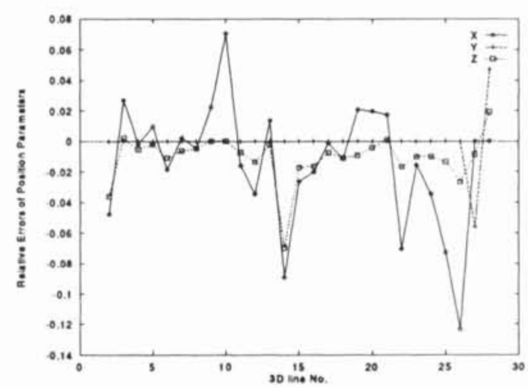

Figure 8: Relative estimating error of position parameters

6 Conclusion

We proposed a new algorithm for estimating parameters of 3D lines from a PSTI. In the method, a 3D STI is scanned to create a PSTI, which has only small part of original data with no much information loss. The loci of 3D lines is extracted by using common edge detectors. Then, the parameters of 3D lines is estimated by recursive recursive optimizing algorithm. In present method, it's not necessary to make the inter-frame corresponding of features, and the limitation of both the slit and camera motion will be decreased. The experiment results illustrate the effectiveness and accuracy of the proposed method. The expansion of this method to deal with real images will be done in future work.

References

[1] C. Tomasi and T. Kanade, "Shape and Motion from Image Streams under Orthography: a Factorization Method", Intern. Journal of Comput. Vision, Vol.9, No.2, pp.137-154(1992).

[2] A. Azarbayejani and P. Pentland, "Recursive Estimation of Motion, Structure, and Focal Length ", IEEE Trans. PAMI, Vol.17, No.6, pp.562-575(1995).

[3] H. Baker and R. Bolles, "Generalizing Epipolar-Plane Image Analysis on the Spatiotemporal Surface", Intern. Journal of Comput. Vision, Vol.?, No.3, pp.33-49(1989).

[4] J. Y. Zheng and S. Tsuji, "Panoramic Representation for Route Recognition by a Mobile Robot ", Intern. Journal of Comput. Vision, Vol.9, No.1, pp.55-76(1992).

[5] T. S. Huang and A. N. Netravali, "Motion and structure from feature correspondences: A Review", IEEE Trans. PAMI, vol.82, no.2, pp.252-268(1994).

[6] C. X. Li, H. T. Zen, and M. Sakauchi, "3D Information Acquisition from Spatiotemporal Image Created by a Hyperbolic Slit", MAV'94, pp.54-57, Kawasaki, Japan(1994).