Ego-Motion Estimation Using Optical Flow Fields Observed from Multiple Cameras

An-Ting Tsao† Yi-Ping Hung‡ Chiou-Shann Fuh† Yong-Sheng Chen†

† Dept. of Computer Science and Information Engineering, National Taiwan University, Taiwan
‡ Institute of Information Science, Academia Sinica, Taipei, Taiwan
Email: hung@iis.sinica.edu.tw

Abstract

In this paper, we consider a multi-camera vision system mounted on a moving object in a static three-dimensional environment. By using the motion flow fields seen by all of the cameras, an algorithm which does not need to solve the point-correspondence problem among the cameras is proposed to estimate the 3D ego-motion parameters of the moving object. Our experiments have shown that using multiple optical flow fields obtained from different cameras can be very helpful for ego-motion estimation.

1 Introduction

Three-dimensional ego-motion estimation has been one of the most important problems for the application of computer vision in mobile robots [10]. Accurate estimation of ego-motion is very helpful for human computer interaction and short-term control such as braking, steering, and navigation.

In the past, there have been many methods [1, 6, 7, 12] which use flow vectors as the basis of their derivations for motion estimation. No matter their derivations are linear or nonlinear, the flow vectors are observed by using single camera. However, there are some drawbacks on using only one camera. First, one can only solve the translation up to the direction, i.e., the absolute scale cannot be determined. This is the well known scaling factor problem. Second, the size of view field substantially affects the accuracy of 3D ego-motion estimation. Third, the solution is not unique, which is the most serious problem. Given a flow field observed from one camera, it may be interpreted as two different kinds of motions.

Let us consider a moving vehicle with two cameras mounted on the left and right sides. Assume there are only two types of motions: one is pure translation toward the front direction and the other is pure rotation around the vertical axis. Now, Suppose we have only the left camera. The flow fields generated by the pure translation and the pure rotation are very similar if the field of view of the camera is not large enough. It is hard to distinguish from this single flow field whether the motion is pure translation or pure rotation.

Next, let us consider the left and right cameras together on this moving vehicle. If there is only translation, the optical flows observed from the two cameras will be the same in scale but *opposite in direction.* If there is only rotation, the optical flows will be the same in both the scale and the direction. Therefore, if we can combine the information contained in the two flow fields appropriately, it will be easier to solve the ambiguity problem.

There are many methods [2, 9, 14] of using multi-camera vision systems to estimate the 3D motion parameters. Most of these systems need to solve the spatial point correspondence problems among all of the cameras. Hence, in a multi-camera vision system, the view fields of the cameras are usually arranged to be overlapping in order to determine the 3D positions of feature points by triangulation. Thus, this kind of approach did not enjoy the fact that the estimation accuracy can be improved by increasing the field of view. Besides, it is not easy to achieve high correction rate when trying to solve the spatial correspondence problem.

We propose to solve the 3D ego-motion estimation problem by a multi-camera system without overlapping view fields. This idea implies: 1) we can obtain a very large view field by using several low-cost small-view-angle cameras, and 2) we do not have to solve the spatial correspondence problem. Based on this idea, we have developed a new algorithm for 3D ego-motion estimation.

In this paper, we consider a multi-camera vision system mounted on a moving object (e.q., human or aircraft) in a static three-dimensional environment. A special case of 3D ego-motion, vehicle-type 3D motion, has been discussed in a previous paper [10]. By us-

1063-6919/97 $10.00 © 1997 IEEE

ing the motion flow fields seen by multiple cameras, an algorithm without solving the spatial point correspondence problem is proposed to estimate 3D ego-motion parameters of the moving object.

2 Ego-Motion Estimation Algorithms

Let us consider an arbitrary multi-camera configuration shown in Fig. 1. Without loss of generality, their focal lengths f_k's (for $k = 1$ to K) are all set to 1. A global coordinate system ($C_g = \{O,\ \boldsymbol{I} = [\mathbf{e}_1|\mathbf{e}_2|\mathbf{e}_3]\}$) attached to the moving object is used to describe the relative position of each camera. The point O is the origin, $\mathbf{e}_1 = (1,0,0)^T$, $\mathbf{e}_2 = (0,1,0)^T$, and $\mathbf{e}_3 = (0,0,1)^T$. Then, the k-th camera coordinate system can be expressed by $C_k = \{\mathbf{b}_k,\ \boldsymbol{R}_k = [\mathbf{u}_{k1}|\mathbf{u}_{k2}|\mathbf{u}_{k3}]\}$. The 3-by-1 vector $\mathbf{b}_k$ denotes the position of O_k, and $\boldsymbol{R}_k$ is a 3-by-3 orthonormal matrix. The positions of all the K cameras are assumed to have been calibrated beforehand.

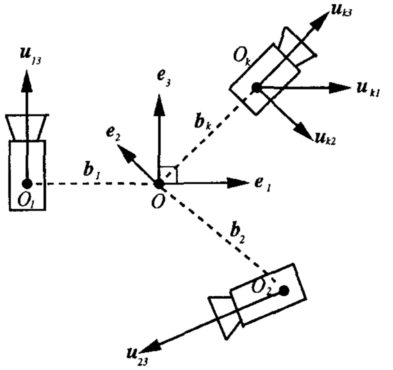

Figure 1: An arbitrary configuration of multiple cameras.

At any time instance, we can compute a flow field for each of the K cameras. The k-th flow field is composed of N_k image feature points and their corresponding flow vectors. Our goal is to compute the 3D ego-motion from the K motion flow fields.

A point P in 3D space can be respectively denoted by $\mathbf{P}$ and $\mathbf{P}_k$ in the two coordinate systems C_g and C_k. These coordinate vectors must satisfy

$$\mathbf{P} \equiv \begin{bmatrix} P_{xk} \\ P_{yk} \\ P_{zk} \end{bmatrix} = \boldsymbol{R}_k \mathbf{P}_k + \mathbf{b}_k. \tag{1}$$

Due to the 3D ego-motion, a point P in the static environment will have an instantaneous 3D motion relative to the global coordinate system C_g [4]:

$$\dot{\mathbf{P}} = -\boldsymbol{\omega} \times \mathbf{P} - \mathbf{t}. \tag{2}$$

where $\boldsymbol{\omega}$ and $\mathbf{t}$ denote the 3D angular velocity and translational velocity of the ego-motion. This 3D relative motion can be expressed with respect to the k-th camera coordinate system C_k:

$$\dot{\mathbf{P}}_k = -\boldsymbol{\omega}_k \times \mathbf{P}_k - \mathbf{t}_k, \tag{3}$$

where

$$\boldsymbol{\omega}_k \equiv \boldsymbol{R}_k^T \boldsymbol{\omega} \text{ and } \mathbf{t}_k \equiv \boldsymbol{R}_k^T [(\boldsymbol{\omega} \times \mathbf{b}_k) + \mathbf{t}]. \tag{4}$$

According to the perspective projection model, the image point $\mathbf{p}_k$ of the 3D point $\mathbf{P}_k$ is

$$\mathbf{p}_k \equiv \begin{bmatrix} p_{xk} \\ p_{yk} \\ 1 \end{bmatrix} = \frac{1}{P_{zk}} \mathbf{P}_k, \tag{5}$$

where P_{zk} is the z-component of $\mathbf{P}_k$. After differentiating both sides of Eq. (5) with respect to time t and substituting Eq. (3) into $\dot{\mathbf{P}}_k$, we have

$$\dot{\mathbf{p}}_k = -(\boldsymbol{\omega}_k \times \mathbf{p}_k) - \frac{\dot{P}_{zk}}{P_{zk}} \mathbf{p}_k - \frac{1}{P_{zk}} \mathbf{t}_k. \tag{6}$$

Applying cross product (×) to both sides of Eq. (6) by $\mathbf{p}_k$, we have

$$\mathbf{p}_k \times [\dot{\mathbf{p}}_k + (\boldsymbol{\omega}_k \times \mathbf{p}_k)] = -\frac{1}{P_{zk}} (\mathbf{p}_k \times \mathbf{t}_k). \tag{7}$$

If we further apply inner product of $\mathbf{t}_k$ to both sides of Eq. (7), we will have the following fundamental equation which does not contain the unknown depth P_{zk}:

$$\{\mathbf{p}_k \times [\dot{\mathbf{p}}_k + (\boldsymbol{\omega}_k \times \mathbf{p}_k)]\} \cdot \mathbf{t}_k = 0. \tag{8}$$

Since we want to estimate $\boldsymbol{\omega}$ and $\mathbf{t}$ using the observations from all the K cameras, the above fundamental equation is re-expressed in terms of $\boldsymbol{\omega}$ and $\mathbf{t}$ by using Eq. (4):

$$\boldsymbol{R}_k \{\mathbf{p}_k \times \left[\dot{\mathbf{p}}_k + (\boldsymbol{R}_k^T \boldsymbol{\omega} \times \mathbf{p}_k)\right]\} \cdot (\boldsymbol{\omega} \times \mathbf{b}_k + \mathbf{t}) = 0. \tag{9}$$

It is Eq. (9) that we use to determine the 3D motion parameters without solving the point correspondence problem.

There are N_k flow vectors associated with the k -th camera. We use $\mathbf{p}_{ki}$ and $\dot{\mathbf{p}}_{ki}$ to represent the i-th point and flow associated with the k -th camera. Eq. (9) can be rewritten to:

$$\mathbf{m}_{ki}^T (\mathbf{h}_k + \mathbf{t}) = 0, \tag{10}$$

where

$$\mathbf{m}_{ki} \equiv \boldsymbol{R}_k \{\mathbf{p}_{ki} \times \left[\dot{\mathbf{p}}_{ki} + (\boldsymbol{R}_k^T \boldsymbol{\omega} \times \mathbf{p}_{ki})\right]\}, \tag{11}$$

$$\mathbf{h}_k \equiv \boldsymbol{\omega} \times \mathbf{b}_k. \tag{12}$$

Algorithm 1: Non-Degenerate Case

According to Eq. (10), we may define an error function J_1' which depends on the unknowns $\boldsymbol{\omega}$ and $\mathbf{t}$:

$$J_1'(\boldsymbol{\omega}, \mathbf{t}) \equiv \sum_{k=1}^{K}\sum_{i=1}^{N_k} \|\mathbf{m}_{ki}^T(\mathbf{h}_k + \mathbf{t})\|^2. \tag{13}$$

We can search for the optimal estimates of $\boldsymbol{\omega}$ and $\mathbf{t}$ by minimizing J_1'. By letting $\partial J_1'/\partial \mathbf{t} = 0$, we have

$$\mathbf{t} = \boldsymbol{M}^{-1}\mathbf{c}, \tag{14}$$

where

$$\boldsymbol{M} \equiv \sum_{k=1}^{K}\sum_{i=1}^{N_k} \mathbf{m}_{ki}\mathbf{m}_{ki}^T \text{ and } \mathbf{c} \equiv -\sum_{k=1}^{K}\sum_{i=1}^{N_k} \mathbf{m}_{ki}\mathbf{m}_{ki}^T\mathbf{h}_k. \tag{15}$$

In the following, the matrix $\boldsymbol{M}$ is sometimes written as $\boldsymbol{M}(\boldsymbol{\omega})$ to emphasize that $\boldsymbol{M}$ is a function of $\boldsymbol{\omega}$.

By substituting Eq. (14) into the $\mathbf{t}$ in Eq. (13), we have a new error function J_1 which only depends on the unknown rotation parameter $\boldsymbol{\omega}$:

$$J_1(\boldsymbol{\omega}) \equiv -\mathbf{c}^T\boldsymbol{M}^{-1}\mathbf{c} + \sum_{k=1}^{K}\sum_{i=1}^{N_k} (\mathbf{m}_{ki}^T\mathbf{h}_k)^2. \tag{16}$$

Therefore the optimal estimate of $\boldsymbol{\omega}$ (denoted by $\hat{\boldsymbol{\omega}}$) is the one that minimizes the error function $J_1(\boldsymbol{\omega})$. Once we have $\hat{\boldsymbol{\omega}}$, the optimal estimate of $\mathbf{t}$, denoted $\hat{\mathbf{t}}$, can be easily obtained by using Eqs. (14) and (15).

Algorithm 2: Degenerate Case

When $\boldsymbol{M}(\boldsymbol{\omega})$ is not full-rank, Eq. (14) fails to solve the corresponding $\mathbf{t}$, which also causes a singularity when calculating J_1.

By considering Eqs. (10) and (14), we know the following conditions will make the matrix $\boldsymbol{M}(\boldsymbol{\omega})$ have a close-to-zero eigenvalue. They include:

(1) $\boldsymbol{\omega}_{true} \approx 0$, i.e., there is almost no rotational motion.

(2) All of the $\mathbf{h}_k$'s are very close to zero. In this case, the whole multi-camera vision system is just like a monocular vision system with K separate view fields, which provides a much larger field of view.

(3) All of the $\mathbf{h}_k$'s are parallel to $\mathbf{t}_{true}$.

Because the optimal solution $\hat{\boldsymbol{\omega}}$ can not be obtained by searching a close-to-singular error function, we have to define a new error function to solve these degenerate cases. In these degenerate cases, we found that all of the $(\mathbf{h}_k + \mathbf{t})$'s are almost parallel to the true translating direction $\mathbf{t}_{n,true}$ ($\equiv \mathbf{t}_{true}/\|\mathbf{t}_{true}\|$). So Eq. (10) can be reduced into the following form:

$$\mathbf{m}_{ki}^T\mathbf{t} = 0, \text{ or } \mathbf{m}_{ki}^T\mathbf{t}_n = 0. \tag{17}$$

The second form of Eq. (17) indicates that only the translating direction $\mathbf{t}_n$ is recoverable in these degenerate cases.

Similarly, we can define an error function J_2' as

$$J_2'(\boldsymbol{\omega}, \mathbf{t}_n) \equiv \sum_{k=1}^{K}\sum_{i=1}^{N_k} (\mathbf{m}_{ki}^T\mathbf{t}_n)^2. \tag{18}$$

Expanding Eq. (18), we have

$$\begin{aligned} J_2'(\boldsymbol{\omega}, \mathbf{t}_n) &= \mathbf{t}_n^T(\sum_{k=1}^{K}\sum_{i=1}^{N_k} \mathbf{m}_{ki}\mathbf{m}_{ki}^T)\mathbf{t}_n \\ &= \mathbf{t}_n^T\boldsymbol{M}\mathbf{t}_n \\ &= \lambda(\mathbf{t}_n)^2 = \lambda, \end{aligned} \tag{19}$$

where the matrix $\boldsymbol{M}$ is defined in Eq. (15), and λ is an eigenvalue of $\boldsymbol{M}(\boldsymbol{\omega})$.

Given an estimate of $\boldsymbol{\omega}$, the best estimate of $\mathbf{t}_n$ should be the eigenvector of $\boldsymbol{M}(\boldsymbol{\omega})$ corresponding to the smallest eigenvalue. If we substitute the above optimal $\mathbf{t}_n$ into J_2', a new error function J_2 which only depends on the unknown $\boldsymbol{\omega}$ can be defined as

$$J_2(\boldsymbol{\omega}) \equiv \text{ the smallest eigenvalue of } \boldsymbol{M}(\boldsymbol{\omega}). \tag{20}$$

Here, the 3-by-3 matrix $\boldsymbol{M}$ is a function of $\boldsymbol{\omega}$.

Therefore, the optimal estimate of $\boldsymbol{\omega}$, denoted by $\hat{\boldsymbol{\omega}}$, is the one which minimizes the error function $J_2(\boldsymbol{\omega})$, and the optimal estimate of $\mathbf{t}_n$ (denoted by $\hat{\mathbf{t}}_n$) is the eigenvector of $\boldsymbol{M}(\hat{\boldsymbol{\omega}})$ corresponding to the smallest eigenvalue.

3 Experimental Results

This section will show some results of real experiments. We use a binocular head (referred to as the IIS head) to simulate a moving object with two cameras mounted on it. The IIS head is built for experiments of active vision, which has four revolute joints and two prismatic joints, as shown in Fig. 2. The two joints on top of the IIS head are for camera verge or gazing. The next two joints below them are for tilting and panning the stereo cameras. All of the above four joints

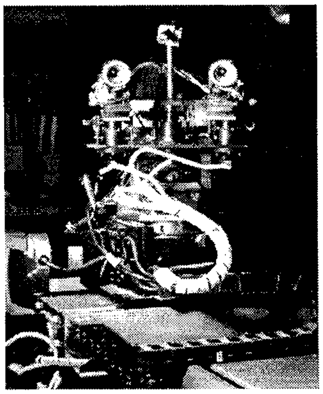

Figure 2: A picture of the IIS head.

are revolute and are mounted on an X-Y table which is composed of two prismatic joints. The lenses of the binocular head are motorized to focus on objects at different distances.

To simplify the coordinate transform, we let the global coordinate system and the left camera coordinate system be identical. Then the left camera coordinate system ($LCCS$) can be expressed by $LCCS = \{\mathbf{b}_l, \boldsymbol{R}_l\}$, where $\mathbf{b}_l = \mathbf{0}$ and $\boldsymbol{R}_l = \boldsymbol{I}$. We let the angle between the optical axes of left and right cameras be about 90°. Notice that the z-axis of $LCCS$ is the same as the optical axis of the left camera, the x-axis points toward the left side of the left camera, and the y-axis points toward the upper side of the left camera. The focal lengths of both cameras are 25 mm, and the fields of view are 15°. The coordinate systems and camera configuration are illustrated in Fig. 3.

3.1 Experiment 1

We let the IIS head move forward, such that the left camera of the IIS head looks ahead and the right camera looks to the right. Table 1 is the true motion parameters used in this experiment. We estimate the ego-motion for three cases: using the left camera only, using the right camera only, and using both the left and right cameras. The scenes viewed from the left and right cameras are shown in Figs. 4(a) and 4(b), respectively. The flow fields observed from the left and right cameras are shown in Figs. 4(c) and 4(d), respectively. Notice that we did not optimize the optical flow, so that some matching errors may occur. Also, the depth of the scene viewed from left camera is in the range from 1.3m to 1.5m, while the depth of the scene viewed from the right camera is about 5m.

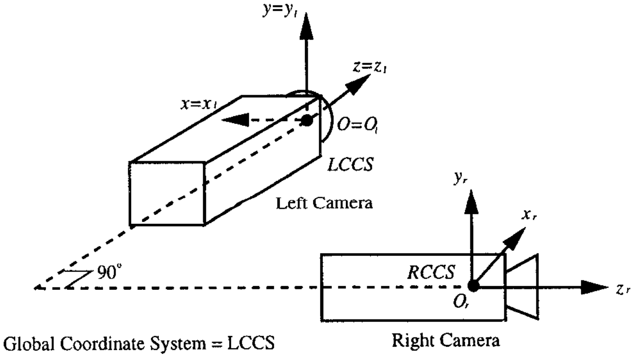

Figure 3: The coordinate systems and camera configuration of the real experiments.

Table 1: True motion parameters used in Experiment 1.

rotation			translation			
ω (deg/frame)			direction			mag.(mm)
ω_x	ω_y	ω_z	t_{xn}	t_{yn}	t_{zn}	$\|\mathbf{t}\|$
0.00	0.00	0.00	-0.017	.045	1.00	20.00

Table 2: Rotational parameters estimated in Experiment 1.

	$\hat{\omega}$ $(deg/frame)$			error
	$\hat{\omega}_x$	$\hat{\omega}_y$	$\hat{\omega}_z$	$\|\hat{\boldsymbol{\omega}} - \boldsymbol{\omega}_{true}\|$
Both	.017	-0.034	.012	.040
Left	0.00	-0.052	-0.052	.073
Right	-0.012	.22	0.00	.22

Table 2 and 3 lists the estimates of the rotational parameters and translational parameters. The results shows that using both cameras performs better than using only one camera. The performance of using only the left camera is also acceptible, which is because the translation direction is close to the optical axis of the left camera [10]. If we use the right camera only, the ambiguity we mentioned in section 1 will occur, where the rotational motion about y-axis can be easily mis-classfied as a translational motion when the field of view is relatively small.

Table 3: Translational parameters estimated in Experiment 1. $\theta(\hat{\mathbf{t}}_n, \mathbf{t}_{n,true})$ is defined as the angle between $\hat{\mathbf{t}}_n$ and $\mathbf{t}_{n,true}$.

	$\mathbf{t}_n$ (translational direction)			error
	$\hat{t}_{xn}$	$\hat{t}_{yn}$	$\hat{t}_{zn}$	$\theta(\hat{\mathbf{t}}_n, \mathbf{t}_{n,true})$
Both	.040	.054	1.00	3.30°
Left	.062	.049	1.00	4.54°
Right	-0.99	-0.051	.10	83.39°

Table 4: True motion parameters used in Experiment 2.

rotation			translation			
ω (deg/frame)			direction			mag.(mm)
ω_x	ω_y	ω_z	t_{xn}	t_{yn}	t_{zn}	$\|\mathbf{t}\|$
.017	.50	-0.023	-0.62	-0.012	-0.78	1.89

Table 5: Rotational parameters estimated in Experiment 2.

	$\hat{\omega}$ $(deg/frame)$			error
	$\hat{\omega}_x$	$\hat{\omega}_y$	$\hat{\omega}_z$	$\|\hat{\omega} - \omega_{true}\|$
Both	0.00	.52	-0.029	.025
Left	.0057	.57	.0057	.086
Right	.017	-0.0057	-0.011	.50

3.2 Experiment 2

In experiment 2, we let the IIS head pan with a small angle. Table 4 is the true motion parameters used in this experiment. Again, we estimate the ego-motion by using the left camera only, the right camera only, and both the left and right camera, respectively. The scenes viewed from the left and right cameras are the same as Figs. 4(a) and 4(b). The flow fields observed from left and right cameras are shown in Fig. 5(a) and 5(b), respectively.

The result of this experiment is given in Tables 5 and 6. As expected, the performance of using both the left and right cameras is the best. The ambiguity problem occurs again when using the right camera only, which is because the depth of the scene viewed from the right camera is as far as five meters and hence the optical flow field looks very similar to the one caused by small pure translation. Notice that the errors of the translational direction are larger than the ones obtained in Experiment 1. This is because the signals of translation in this experiment are very small. Whenever there is noise, the signals of translation will be seriously corrupted, and it is hard to estimate the translational parameters with high accuracy.

4 Conclusion

In this paper, we propose a method for ego-motion estimation using a multiple-camera vision system. By combining the information contained in the multiple optical flows observed from different cameras, some ambiguity problems can be avoided and the accuracy can be improved. Our algorithm considers two cases separately: non-degenerate case and degenerate case.

Table 6: Translational parameters estimated in Experiment 2. $\theta(\hat{\mathbf{t}}_n, \mathbf{t}_{n,true})$ is defined as the angle between $\hat{\mathbf{t}}_n$ and $\mathbf{t}_{n,true}$.

	$\mathbf{t}_n$ (translational direction)			error
	$\hat{t}_{xn}$	$\hat{t}_{yn}$	$\hat{t}_{zn}$	$\theta(\hat{\mathbf{t}}_n, \mathbf{t}_{n,true})$
Both	-0.052	-0.034	-1.00	36.00
Left	-0.049	-0.055	-1.00	36.00
Right	0.083	.99	-0.095	89.00

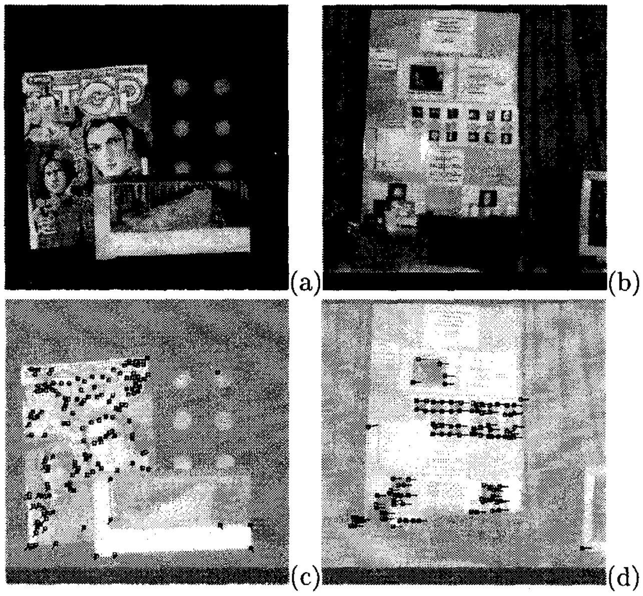

Figure 4: The images and the flow fields used in experiment 1: (a) The scene viewed from the left camera. (b) The scene viewed from the right camera. (c) The optical flow field obtained from the left camera. (d) The optical flow field obtained from the right camera. The flow vectors in the figures are enlarged by a factor of two.

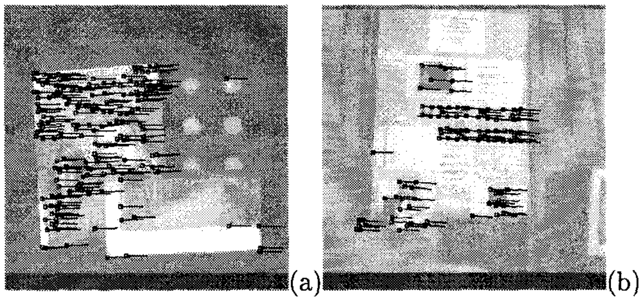

Figure 5: The flow fields used in experiment 2: (a) The optical flow field obtained from the left camera. (b) The optical flow field obtained from the right camera. The flow vectors in the figures are also enlarged by a factor of two.

One potential application of our multiple-camera approach is the "inside-out" (or "outward looking") head tracker for virtual reality. The current outward looking head tracker requires structured environments, e.g. regular pattern in the ceiling. Our approach does not require specially-designed environment, as long as the environment have enough features for computing optical flow.

Different camera configurations have different performance on ego-motion estimation. In this paper, we have not analyzed the performance between different camera configurations. Some analysis on finding the optimal camera configuration can be found in another paper [10].

Acknowledgements

This work was supported in part by the National Science Council of Taiwan, under Grants NSC 86-2745-E-001-007, NSC 86-2212-E-002-025, and by Mechanical Industry Research Laboratories, Industrial Technology Research Institute, under Grant MIRL 863K67BN2.

References

[1] G. Adiv, "Inherent Ambiguities in Recovering 3D Motion and Structure from a Noisy Flow Field," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 11, pp. 477-489, 1989.

[2] N. Ayache and F. Lustman, "Trinocular Stereo Vision for Robotics," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 13, pp. 73-85, 1991.

[3] T. Broida and R. Chellappa, "Estimation of Object Motion Parameters from Noisy Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 8, pp. 90-99, 1986.

[4] T. Broida and R. Chellappa, "Estimating the Kinematics and Structure of a Rigid Object from a Sequence of Monocular Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 13, pp. 497-512, 1991.

[5] W. Burger and B. Bhanu, "Estimating 3D Egomotion from Perspective Image Sequences," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, pp. 1040-1058, 1990.

[6] D. J. Heeger and A. D. Jepson, "Subspace Methods for Recovering Rigid Motion I: Algorithm and Implementation," *International Journal of Computer Vision*, Vol. 7, pp. 95-117, 1992.

[7] R. Hummel and V. Sundareswaran, "Motion Parameter Estimation from Global Flow Field Data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 15, pp. 459-476, 1993.

[8] M. K. Leung and T. S. Huang, "Estimating Three-Dimensional Vehicle Motion in an Outdoor Scene Using Stereo Image Sequences," *International Journal of Imaging Systems and Technology*, Vol. 4, pp. 80-97, 1992.

[9] L. Li and J. H. Duncan, "3D Translational Motion and Structure from Binocular Image Flows," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 15, pp. 739-745, 1993.

[10] L. G. Liou and Y. P. Hung, "Vehicle-Type Ego-Motion Estimation Using Multiple Cameras," *Proceedings of Asian Conference on Computer Vision*, pp. 166-170, Singapore, 1995.

[11] Y. Liu and T. S. Huang, "Vehicle-Type Motion Estimation from Multi-frame Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 15, pp. 802-808, 1993.

[12] K. Prazdny, "Determining the Instantaneous Direction of Motion from Optical Flow Generated by a Curvilinearly Moving Observer," *Computer Graphics and Image Processing*, Vol. 17, pp. 238-248, 1981.

[13] T. Vieville, E. Clergue, and P. E. D. S. Facao, "Computation of Ego-Motion and Structure from Visual and Inertial Sensors Using the Vertical Cue," *Proceedings of International Conference on Computer Vision*, pp. 591-598, Berlin, Germany, 1993.

[14] J. Weng, P. Cohen, and N. Rebibo, "Motion and Structure Estimation from Stereo Image Sequence," *IEEE Transactions on Robotics and Automation*, Vol. 8, pp. 362-382, 1992.