

# Translation estimation for single viewpoint cameras using lines

Sang Ly<sup>1</sup>, Cédric Demonceaux<sup>1</sup> and Pascal Vasseur<sup>1,2</sup>

<sup>1</sup>Laboratory MIS. University of Picardie Jules Verne. Amiens. France

<sup>2</sup>Laboratory Heudiasyc. University of Technology of Compiègne. France

E-mail addresses: {sang.ly, cedric.demonceaux, pascal.vasseur}@u-picardie.fr

**Abstract**—We present a translation estimation method for single viewpoint (SVP) cameras using line features. Images captured by multiple central cameras such as perspective, central catadioptric and fisheye cameras are mapped to spherical images using the unified projection model. It is possible to recover the camera rotations using vanishing points of parallel line sets. We then estimate the translations from known rotations and line images on the spheres. The algorithm has been validated on simulated data and real images. This vision-based estimation approach can be applied in navigation of autonomous robots besides the conventional devices such as Global Positioning System (GPS) and Inertial Navigation System (INS). It helps vision-based localization of a single robot or recovery of relative positions among multiple robots equipped with different types of cameras.

## I. INTRODUCTION

Localization is a critical issue in robot motion control. It is possible to acquire robot motion data using various types of sensors. While a GPS is sensitive to signal dropout and an INS may accumulate the localization error over time, a vision-based approach for position estimation is a suitable choice. That is one of the reasons why cameras are widely used in robot guidance. There exist different methods for camera motion estimation which may be organized into three main categories [5]: 1. egomotion estimation using optical flow, 2. decomposition of fundamental or essential (F/E) or homography (H) matrices to obtain camera motion and 3. two-stage estimation, in other words decoupling of rotation and translation. This paper presents an approach of translation estimation for all single projection center cameras supposing that rotations among them have been known. We therefore focus on methods belonging to the last category.

### A. Perspective vision

Multiple view reconstruction methods may be started with *factorization* techniques. First, Tomasi and Kanade [26] proposed a factorization method to recover the scene structure and camera motion from a sequence of images. The implementation of this method is simple and provides reliable results. However, its use is limited to affine camera model and it requires that all point features are visible in all images [10]. The projective factorization, an extension of the previous one to projective camera model, was developed in [25], [11], [19]. These approaches again require the presence of all points in all frames and they may not converge to correct solution in all cases [10]. Martinec and Pajdla [21] improved Jacobs' method [13] to deal with data missing

problem, which occurs when point features disappear in some frames.

Reconstruction problems can also be solved by *bundle adjustment* [29]. Although this iterative technique may be applied in a wide class of optimization problems, it cannot be proven to converge to optimal solution from an arbitrary initial point [10]. Therefore, an initialisation technique, such as 8-point algorithm [18], [9], [28] is usually employed in order to provide a good starting point for bundle adjustment. The main disadvantage of bundle adjustment is that it is a slow technique [10] and not robust to significant measurement noise [15].

Recently,  $L_\infty$  optimization methods have been proposed to solve the structure and motion problem. In [14], Kahl presented an  $L_\infty$  approach based on second-order cone programming (SOCP) to estimate the camera translations and 3D points assuming known rotations. This technique permits an efficient computation of global estimates for a wide range of geometric vision problems. Moreover, its solutions are invariant to projective and similarity transformations. Sim and Hartley [24] recovered the camera translations also using  $L_\infty$  minimization based on SOCP. Martinec and Pajdla [22] solved the reconstruction problem in two stages: estimate first the camera rotations linearly in least squares and then the camera translations using SOCP. A similar technique for quasi-convex optimization was developed in [16]. The main disadvantage of  $L_\infty$ -norm is that it is not robust to outliers [15]. Method proposed in [14] may fail due to a single wrong correspondence [22].

### B. Omnidirectional vision

Omnidirectional vision systems possess a wider field of view than conventional cameras. Such devices can be built up from an arrangement of several cameras or a single camera with special lenses such as fisheye or with mirrors of particular curvatures. In structure and motion problem, omnidirectional sensors using a single camera play an important role as they overcome several disadvantages when working with perspective cameras, such as translation/rotation ambiguity, lack of features and the large number of views in use.

In [1], Antone and Teller first estimated camera rotations using vanishing points calculated from parallel line sets in 3D scene and then extracted camera translations using Hough transform. This method provided interesting results but might be time consuming. Moreover, two stages of their algorithm require different feature types, i.e. lines for rotation and

points for translation estimations. Kim and Hartley presented a translation estimation from omnidirectional images assuming known rotations [17]. The translations along multiple views were recovered from point correspondences using a constrained minimization. In [20], Makadia and others proposed a 3D motion computation from two omnidirectional views without correspondences. The camera rotation and translation were estimated using the Fourier transform of the spherical images. Although this approach is robust to wrong feature detection and outliers, it is computational expensive and sensitive to dynamic environment [5]. Bazin et al. presented a motion estimation approach also by decoupling rotation and translation [5]. The relative orientation of two para-catadioptric cameras was determined using vanishing points calculated from parallel lines. Then, the translation was recovered from known rotation and two point correspondences. Again, this technique requires different feature types, i.e. lines and points, for two stages.

In this paper, we present a translation estimation method for SVP cameras assuming known rotations, in which:

1. We use the unified projection model proposed by Mei [23], a slightly modified version of the models developed by Geyer [7] and Barreto [2]. This model encompasses a large range of central projection devices including fisheye lenses [30]. Therefore, our method may be applied to perspective, central catadioptric and fisheye cameras. To the best of our knowledge, such translation estimation approach for a wide class of central imaging systems has not been presented.
2. Lines are used as the primitive feature in our approach for several reasons: Such features are typically more stable than points and are less likely to be produced by clutter or noise, especially in man-made environment [6]. Compared to point features, lines are less numerous but more informative, they have geometrical and topological characteristics which are useful for matching [8], [3]. Moreover, the authors in [1], [5] recovered rotations from lines and translations from points. We propose to use a single type of feature, i.e. lines for both rotation and translation estimations, that may help optimise the computation time of such two-stage technique. A fast motion recovery is obviously useful in robotic applications.

In the following section, we develop the multi-view geometry for single viewpoint cameras. Next, we present our translation estimation using line feature. We show then the experimental results from simulated data and real images before the conclusions.

## II. MULTI-VIEW GEOMETRY FOR SINGLE VIEWPOINT CAMERAS

Central imaging systems including fisheye lenses can be modelled by the unitary sphere, therefore they are considered to be equivalent to spherical cameras. Noting that line correspondences can be used only with more than two views [10], we develop three- and four-view geometry for spherical cameras, an extension of the two- and three-view geometry

derived by Torii et al. [27], in which they demonstrated the bilinear and trilinear constraints for spherical cameras, but did not discuss its further application.

**Notation:** Matrices are denoted using Sans Serif font, vectors in bold font and scalars in italics.

Consider four spherical cameras with projection centers  $C_i$  ( $i=1..4$ ) as illustrated in figure 1. A line  $L$  in 3D space is projected to spherical images as great circles  $l_i$ , which have the corresponding normals  $\mathbf{n}_i$ .  $L$  can be expressed vectorially by  $L = X_0 + \mu \mathbf{d}$  where  $L, X_0, \mathbf{d} \in \mathbb{R}^3$  and  $\mu \in \mathbb{R}$ .  $\mathbf{n}_i \in \mathbb{R}^3$  are normal correspondences in four spherical images.

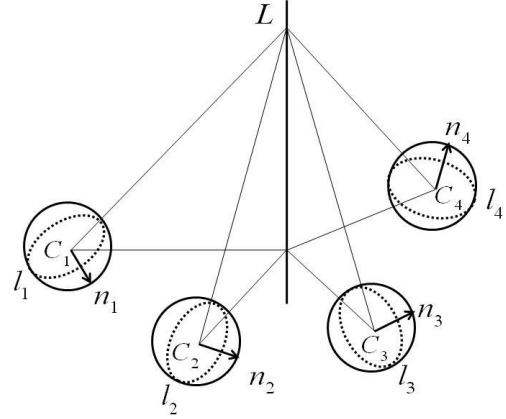


Fig. 1. Four view geometry of spherical cameras

Assuming that  $C_1$  is at the origin of our coordinate system, let  $[R_2|t_2]$ ,  $[R_3|t_3]$  and  $[R_4|t_4]$  be the  $[Rotation|translation]$  between  $(C_1$  and  $C_2)$ ,  $(C_1$  and  $C_3)$  and  $(C_1$  and  $C_4)$  respectively.

As the line  $L$  lies on the projective planes passing through great circles  $l_i$  and perpendicular to normals  $\mathbf{n}_i$ , we have the following relations in which  $L$  is expressed in  $C_1$  and  $\mathbf{n}_i$  are expressed in  $C_i$  ( $i=1..4$ ):

$$\mathbf{n}_1^T L = 0 \quad (1)$$

$$\mathbf{n}_2^T (R_2 L + t_2) = 0 \quad (2)$$

$$\mathbf{n}_3^T (R_3 L + t_3) = 0 \quad (3)$$

$$\mathbf{n}_4^T (R_4 L + t_4) = 0 \quad (4)$$

### A. Three-view geometry

Each triplet, i.e. group of three equations, chosen among four equations (1) to (4) expresses the relation among the line  $L$  and the corresponding normals in three views. For example, the relation among three views 1,2 and 3 consists of equations (1),(2) and (3) and can also be represented as follows:

$$A \hat{L} = 0 \quad (5)$$

where

$$A = \begin{bmatrix} \mathbf{n}_1^T & 0 \\ \mathbf{n}_2^T R_2 & \mathbf{n}_2^T t_2 \\ \mathbf{n}_3^T R_3 & \mathbf{n}_3^T t_3 \end{bmatrix} \quad (6)$$

and  $\hat{\mathbf{L}} = (\mathbf{L}^T, 1)^T$ .

The existence of at least a non-zero solution in (5) requires that the 3x4 matrix  $\mathbf{A}$  has rank 2. It follows that there is a linear dependence among three rows of  $\mathbf{A}$ . Denoting  $\mathbf{A} = (\mathbf{a}_1^T, \mathbf{a}_2^T, \mathbf{a}_3^T)$ , the linear relation can be written as  $\mathbf{a}_1 = \alpha \mathbf{a}_2 + \beta \mathbf{a}_3$ . Noting that  $a_{14} = 0$ , we can select  $\alpha = k \mathbf{t}_3^T \mathbf{n}_3$  and  $\beta = -k \mathbf{t}_2^T \mathbf{n}_2$  for some scalar  $k$ . This can be applied to the first three columns of  $\mathbf{A}$  to obtain the next relation:

$$\mathbf{n}_1^T = \alpha \mathbf{n}_2^T \mathbf{R}_2 + \beta \mathbf{n}_3^T \mathbf{R}_3$$

$$\mathbf{n}_1 = \alpha \mathbf{R}_2^T \mathbf{n}_2 + \beta \mathbf{R}_3^T \mathbf{n}_3$$

$$\mathbf{n}_1 = k \mathbf{t}_3^T \mathbf{n}_3 \mathbf{R}_2^T \mathbf{n}_2 - k \mathbf{t}_2^T \mathbf{n}_2 \mathbf{R}_3^T \mathbf{n}_3$$

Views 1, 2 and 3:

$$\mathbf{R}_2^T \mathbf{n}_2 \mathbf{n}_3^T \mathbf{t}_3 - \mathbf{R}_3^T \mathbf{n}_3 \mathbf{n}_2^T \mathbf{t}_2 + \hat{k} \mathbf{n}_1 = 0 \quad (7)$$

with scalar  $\hat{k} = -1/k$ . Note that  $k$  is definitely different from zero.

In the same manner, we can establish the relations in the other triplets:

Views 1, 2 and 4:

$$\mathbf{R}_2^T \mathbf{n}_2 \mathbf{n}_4^T \mathbf{t}_4 - \mathbf{R}_4^T \mathbf{n}_4 \mathbf{n}_2^T \mathbf{t}_2 + \tilde{k} \mathbf{n}_1 = 0 \quad (8)$$

Views 1, 3 and 4:

$$\mathbf{R}_3^T \mathbf{n}_3 \mathbf{n}_4^T \mathbf{t}_4 - \mathbf{R}_4^T \mathbf{n}_4 \mathbf{n}_3^T \mathbf{t}_3 + \check{k} \mathbf{n}_1 = 0 \quad (9)$$

for some scalars  $\tilde{k}, \check{k}$ .

Each equation (7), (8) or (9) relates the normal correspondences in a triplet of views to each other through the transformations among those views.

### B. Four-view geometry

Equations (1) to (4) can be arranged in a linear system as follows:

$$\mathbf{B} \hat{\mathbf{L}} = 0 \quad (10)$$

where

$$\mathbf{B} = \begin{bmatrix} \mathbf{n}_1^T & 0 \\ \mathbf{n}_2^T \mathbf{R}_2 & \mathbf{n}_2^T \mathbf{t}_2 \\ \mathbf{n}_3^T \mathbf{R}_3 & \mathbf{n}_3^T \mathbf{t}_3 \\ \mathbf{n}_4^T \mathbf{R}_4 & \mathbf{n}_4^T \mathbf{t}_4 \end{bmatrix} \quad (11)$$

and  $\hat{\mathbf{L}} = (\mathbf{L}^T, 1)^T$ .

Again, the system (10) has at least a non-zero solution when the 4x4 matrix  $\mathbf{B}$  is not invertible. It follows that the determinant of this square matrix is null, that is expressed in the next equation:

$$\begin{vmatrix} \mathbf{n}_1^T & & & & \\ \mathbf{n}_2^T \mathbf{R}_2 & \mathbf{n}_2^T \mathbf{t}_2 & & & \\ \mathbf{n}_3^T \mathbf{R}_3 & \mathbf{n}_3^T \mathbf{t}_3 & & & \\ \mathbf{n}_4^T \mathbf{R}_4 & \mathbf{n}_4^T \mathbf{t}_4 & & & \end{vmatrix} = 0 \quad (12)$$

where  $|\cdot|$  denotes the matrix determinant.

Equation (12) relates the normal correspondences in a quadruplet of views to each other through the transformations among those views.

## III. TRANSLATION ESTIMATION METHOD

In this section, we present a method to estimate the translations  $\mathbf{t}_2$ ,  $\mathbf{t}_3$  and  $\mathbf{t}_4$  among the SVP cameras from known rotations  $\mathbf{R}_2$ ,  $\mathbf{R}_3$  and  $\mathbf{R}_4$  and line/normal correspondences  $\mathbf{n}_i$  ( $i=1..4$ ) in spherical images.

### A. Translation estimation from triplets of views

It is possible to estimate the translations  $\mathbf{t}_2$  and  $\mathbf{t}_3$  from (7),  $\mathbf{t}_2$  and  $\mathbf{t}_4$  from (8),  $\mathbf{t}_3$  and  $\mathbf{t}_4$  from (9). Hence, we can concatenate (7), (8) and (9) in a linear system that permits the estimation of all translations.

$$\begin{cases} \mathbf{R}_2^T \mathbf{n}_2 \mathbf{n}_3^T \mathbf{t}_3 - \mathbf{R}_3^T \mathbf{n}_3 \mathbf{n}_2^T \mathbf{t}_2 + \hat{k} \mathbf{n}_1 = 0 \\ \mathbf{R}_2^T \mathbf{n}_2 \mathbf{n}_4^T \mathbf{t}_4 - \mathbf{R}_4^T \mathbf{n}_4 \mathbf{n}_2^T \mathbf{t}_2 + \tilde{k} \mathbf{n}_1 = 0 \\ \mathbf{R}_3^T \mathbf{n}_3 \mathbf{n}_4^T \mathbf{t}_4 - \mathbf{R}_4^T \mathbf{n}_4 \mathbf{n}_3^T \mathbf{t}_3 + \check{k} \mathbf{n}_1 = 0 \end{cases} \Leftrightarrow \mathbf{M} \mathbf{X} = 0 \quad (13)$$

where

$$\mathbf{M} = \begin{bmatrix} -\mathbf{R}_3^T \mathbf{n}_3 \mathbf{n}_2^T & \mathbf{R}_2^T \mathbf{n}_2 \mathbf{n}_3^T & 0 & \mathbf{n}_1 & 0 & 0 \\ -\mathbf{R}_4^T \mathbf{n}_4 \mathbf{n}_2^T & 0 & \mathbf{R}_2^T \mathbf{n}_2 \mathbf{n}_4^T & 0 & \mathbf{n}_1 & 0 \\ 0 & -\mathbf{R}_4^T \mathbf{n}_4 \mathbf{n}_3^T & \mathbf{R}_3^T \mathbf{n}_3 \mathbf{n}_4^T & 0 & 0 & \mathbf{n}_1 \end{bmatrix} \quad (14)$$

and

$$\mathbf{X} = (\mathbf{t}_2^T, \mathbf{t}_3^T, \mathbf{t}_4^T, \hat{k}, \tilde{k}, \check{k})^T \quad (15)$$

with  $\hat{k}$ ,  $\tilde{k}$  and  $\check{k}$  being some scalars in triplets of views  $\{1,2,3\}$ ,  $\{1,2,4\}$  and  $\{1,3,4\}$  respectively.

We may notice that two triplets already permit the estimation of  $\mathbf{t}_2$ ,  $\mathbf{t}_3$  and  $\mathbf{t}_4$ . However, we use all three triplets as they are independent of each other. Obviously, from the last three columns of matrix  $\mathbf{M}$  in (14), it is impossible that one triplet is dependent on the others.

Therefore, given a line/normal correspondence  $\mathbf{n}_i$  in four spherical views, (13) gives us a linear system in the translations  $\mathbf{t}_2$ ,  $\mathbf{t}_3$  and  $\mathbf{t}_4$  and three scalars. Each extra correspondence enlarges the matrix  $\mathbf{M}$  by 9 lines and 3 columns and the unknown vector  $\mathbf{X}$  by 3 scalars.

With  $N$  correspondences, we have the following system:

$$\hat{\mathbf{M}} \hat{\mathbf{X}} = 0 \quad (16)$$

where  $\hat{\mathbf{M}}$  is a  $9N \times (9+3N)$  matrix and  $\hat{\mathbf{X}}$  a  $(9+3N)$ -vector. They are established as follows:

$$\hat{\mathbf{M}} = [\hat{\mathbf{M}}_1, \hat{\mathbf{M}}_2] \quad (17)$$

with  $\hat{\mathbf{M}}_1 =$

$$\begin{bmatrix} -\mathbf{R}_3^T \mathbf{n}_{31} \mathbf{n}_{21}^T & \mathbf{R}_2^T \mathbf{n}_{21} \mathbf{n}_{31}^T & 0 \\ -\mathbf{R}_4^T \mathbf{n}_{41} \mathbf{n}_{21}^T & 0 & \mathbf{R}_2^T \mathbf{n}_{21} \mathbf{n}_{41}^T \\ 0 & -\mathbf{R}_4^T \mathbf{n}_{41} \mathbf{n}_{31}^T & \mathbf{R}_3^T \mathbf{n}_{31} \mathbf{n}_{41}^T \\ \dots & \dots & \dots \\ \dots & \dots & \dots \\ -\mathbf{R}_3^T \mathbf{n}_{3N} \mathbf{n}_{2N}^T & \mathbf{R}_2^T \mathbf{n}_{2N} \mathbf{n}_{3N}^T & 0 \\ -\mathbf{R}_4^T \mathbf{n}_{4N} \mathbf{n}_{2N}^T & 0 & \mathbf{R}_2^T \mathbf{n}_{2N} \mathbf{n}_{4N}^T \\ 0 & -\mathbf{R}_4^T \mathbf{n}_{4N} \mathbf{n}_{3N}^T & \mathbf{R}_3^T \mathbf{n}_{3N} \mathbf{n}_{4N}^T \end{bmatrix} \quad (18)$$

$$\hat{\mathbf{M}}_2 =$$

$$\text{diag}(\mathbf{n}_{11}, \mathbf{n}_{11}, \mathbf{n}_{11}, \mathbf{n}_{12}, \dots, \mathbf{n}_{1(N-1)}, \mathbf{n}_{1N}, \mathbf{n}_{1N}, \mathbf{n}_{1N}) \quad (19)$$

$$\hat{\mathbf{X}} = (\mathbf{t}_2^T, \mathbf{t}_3^T, \mathbf{t}_4^T, \hat{k}_1, \tilde{k}_1, \check{k}_1, \dots, \hat{k}_N, \tilde{k}_N, \check{k}_N)^T \quad (20)$$

where

- $\mathbf{n}_{ij}$  is the  $j$ th correspondence in the  $i$ th view ( $i=1..4$ ,  $j=1..N$ )
- $\hat{k}_j$ ,  $\tilde{k}_j$  and  $\check{k}_j$  are respectively some scalars in three triplets  $\{1,2,3\}$ ,  $\{1,2,4\}$  and  $\{1,3,4\}$  for the  $j$ th correspondence

### B. Translation estimation from quadruplet of views

Equation (12) describes the geometry constraint of a line/normal correspondence  $\mathbf{n}_i$  in four views.  $N$  correspondences provide a linear system in the translations  $\mathbf{t}_2$ ,  $\mathbf{t}_3$  and  $\mathbf{t}_4$  as shown below:

$$\mathbf{Q} \begin{pmatrix} \mathbf{t}_2 \\ \mathbf{t}_3 \\ \mathbf{t}_4 \end{pmatrix} = 0 \quad (21)$$

where

$$\mathbf{Q} = \begin{bmatrix} \Delta_{21} & \Delta_{31} & \Delta_{41} \\ \Delta_{22} & \Delta_{32} & \Delta_{42} \\ \dots & \dots & \dots \\ \Delta_{2N} & \Delta_{3N} & \Delta_{4N} \end{bmatrix} \quad (22)$$

$$\Delta_{2j} = \begin{vmatrix} \mathbf{n}_{1j}^T \mathbf{R}_3 \\ \mathbf{n}_{3j}^T \mathbf{R}_3 \\ \mathbf{n}_{4j}^T \mathbf{R}_4 \end{vmatrix} \mathbf{n}_{2j}^T, \Delta_{3j} = - \begin{vmatrix} \mathbf{n}_{1j}^T \mathbf{R}_2 \\ \mathbf{n}_{2j}^T \mathbf{R}_2 \\ \mathbf{n}_{4j}^T \mathbf{R}_4 \end{vmatrix} \mathbf{n}_{3j}^T \text{ and} \quad (23)$$

$$\Delta_{4j} = \begin{vmatrix} \mathbf{n}_{1j}^T \mathbf{R}_2 \\ \mathbf{n}_{2j}^T \mathbf{R}_2 \\ \mathbf{n}_{3j}^T \mathbf{R}_3 \end{vmatrix} \mathbf{n}_{4j}^T$$

with  $\mathbf{n}_{ij}$  being the  $j$ th correspondence in the  $i$ th view ( $i=1..4$ ,  $j=1..N$ ).

Using a linear approach such as Single Value Decomposition (SVD), we can solve the systems (16) or (21) to recover the translations among four cameras.

Ignoring the effects of noise in the normals and rotations, the rank of the systems (16) and (21) generated from  $N \geq 2$  line/normal correspondences is analyzed in order to study the theoretical number of correspondences sufficient for the estimation. The result is given in the following table. Noting that  $\hat{\mathbf{M}}$  is a  $9N \times (9+3N)$  matrix, thus four correspondences or more permit the estimation of three translations using three triplets in section III.A. Matrix  $\mathbf{Q}$  is of size  $N \times 9$ , therefore the translations can be estimated using the quadruplet in section III.B from at least eight correspondences.

$N$	$\text{rank}(\hat{\mathbf{M}})$	$\text{rank}(\mathbf{Q})$
2	10	$\min(N,9)$
3	15	
more than 4	$8+3N$	

## IV. EXPERIMENTAL RESULTS

### A. Synthetic data

Since the proposed method is based on line projections in spherical images, we first create 3D lines surrounding the centers  $\mathbf{C}_i$  ( $i=1..4$ ) of four spherical cameras. The average baseline among four cameras is 1000 mm and the average distance of the 3D lines from the cameras is 15000 mm. These lines are mapped to the spherical images as great circles and corresponding normals. The translations  $\mathbf{t}_2$ ,  $\mathbf{t}_3$  and  $\mathbf{t}_4$  among four cameras are recovered from the *line/normal correspondences* together with known rotations  $\mathbf{R}_2$ ,  $\mathbf{R}_3$  and  $\mathbf{R}_4$  using 3-triplet and quadruplet approaches presented in section III.

To evaluate the results, we also implemented the translation estimation approach proposed by Kim and Hartley in [17]. In [17], the translations among spherical cameras are estimated from *point correspondences* and known rotations using a constrained minimization.

Normals in our method and points in Kim-Hartley [17] are on unitary spheres, thus may be specified by elevation and azimuth angles. Gaussian noise of zero mean and varying standard deviations (0.1 and 0.5 degrees) is added to two angles of every normal and every point. To simulate the inaccuracy in preliminary rotation estimation, the roll, pitch and yaw angles of each rotation are perturbed by Gaussian noise of zero mean and standard deviations from 0.0 to 1.0 degrees. Figure 2 shows the average angular error of three translations after 1000 runs.

It can be seen that using triplets of views gives better estimation than using a quadruplet of views. The reason is that matrix  $\mathbf{Q}$  in (21) is badly conditioned. Its elements are very near to zero. For example with one correspondence,  $\Delta_{4j}$  in (23) is composed of the trilinearity of views 1, 2 and 3. The determinant  $|\cdot|$  in  $\Delta_{4j}$  is exactly the determinant of the  $3 \times 3$  minor of  $\mathbf{A}$  in (6). Compared to Kim-Hartley [17], our estimation from three triplets of views provides nearly similar accuracy. However, we may notice that using lines is more favorable than using points in the feature detection and matching phases, notably when we used different types of cameras. In addition, adding the same noise has an influence to normals more critical than to points.

### B. Real images

We show in this subsection the translation estimation using images captured by perspective, para-catadioptric and fisheye cameras. We placed our cameras at four different positions and recovered their configuration from the captured images. The estimation requires line correspondences among four views and relative orientations among four camera positions, therefore we also discuss here the preliminary steps:

1. *Spherical projection*: Original images are mapped to spherical images using the unified projection model in [23].
2. *Line detection*: A fast central catadioptric line extraction method is proposed in [4]. The extraction is composed of a splitting step and a merging step in both original

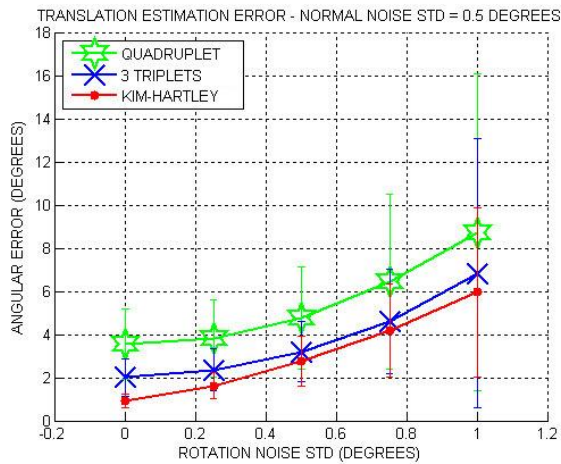
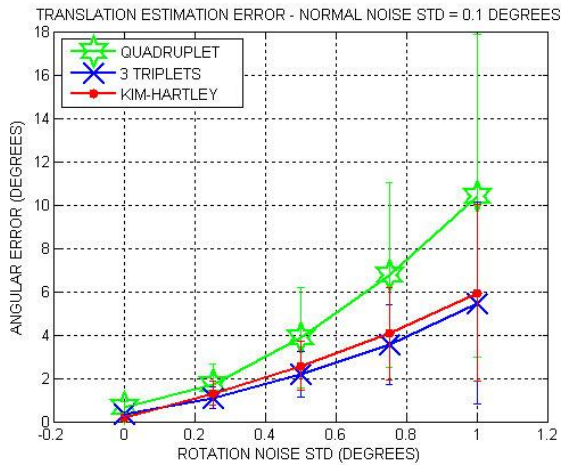


Fig. 2. Translation estimation errors from quadruplet, 3-triplet and Kim-Hartley [17] approaches

and spherical images. Modifying the projection model, we extend this approach to a line detection algorithm applicable to a wide range of SVP cameras.

3. *Line matching*: To the best of our knowledge, a method for matching lines across multiple views captured by cameras of different classes has not been developed in state of the art. To focus on our translation estimation that requires just a few number of line correspondences, line matching has been done offline and manually. Examples of line matching among images taken by fisheye and para-catadioptric cameras are illustrated in figure 3.
4. *Rotation estimation*: Rotation between two views can be estimated using the correspondences of two vanishing points [5]. We first detect vanishing points from bundles of parallel lines in each view and then recover the rotation from corresponding vanishing points using the solution proposed by Horn in [12].

In figure (4), we show the position recovery for a fisheye camera using the 3-triplet, quadruplet and Kim-Hartley [17]

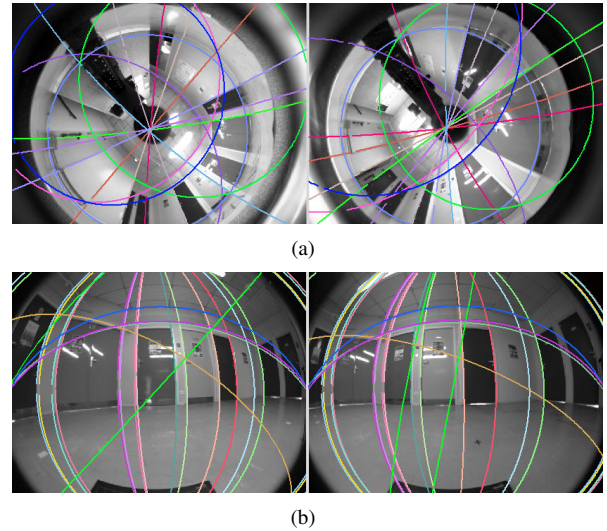


Fig. 3. Line matching among images captured by para-catadioptric (a) and fisheye (b) cameras. Corresponding lines are plotted with same color

methods. We used 19 line/normal correspondences in our approach and 19 point correspondences in Kim-Hartley [17]. The ratio among the translations ( $C_3C_4 : C_4C_2 : C_2C_1$ ) is used to evaluate these methods and summarized in the next table. The 3-triplet estimation provides the best result. However, it should be noticed that the estimation results depend much on the line detection in our method and point detection in Kim-Hartley [17].

	$(C_3C_4 : C_4C_2 : C_2C_1)$
Ground truth	(3.00 : 4.00 : 4.00)
3-triplet	(3.00 : 4.03 : 4.02)
Quadruplet	(3.00 : 3.93 : 3.94)
Kim-Hartley [17]	(3.00 : 4.06 : 4.02)

In figure (5) is the position reconstruction for perspective and para-catadioptric cameras using the 3-triplet approach. For the perspective camera (on the left column), the ground truth is  $(C_3C_4 : C_4C_1 : C_1C_2) = (3.00 : 3.00 : 3.00)$  and the reconstruction from 13 line/normal correspondences is  $(3.00 : 2.99 : 3.01)$ . For the para-catadioptric camera (on the right column), the ground truth is  $(C_3C_4 : C_4C_2 : C_2C_1) = (3.00 : 4.00 : 4.00)$  and the recovery from 14 line/normal correspondences is  $(3.00 : 4.18 : 4.12)$ .

## V. CONCLUSIONS AND FUTURE WORKS

We presented in this paper a linear approach of translation estimation for imaging systems equivalent to the central projection model. Assuming known rotations, the translations among cameras of various types can be recovered using lines. We validated our method using simulated data and real images, and compared to the translation estimation using point features developed by Kim-Hartley [17]. Using the unified projection model, this approach can be applied to a wide class of central imaging systems. Moreover, line features are employed to overcome the disadvantages of point features when working with images taken from dissimilar categories

of cameras. To estimate the translations among four cameras, the 3-triplet is more favorable than the quadruplet approach due to the robustness of the mathematical system used to solve the problem. The proposed method promises a fast and robust motion recovery which is very important and helpful in autonomous robotics. We are applying this estimation to a hybrid stereo vision system mounted on an autonomous robot. Such hybrid stereo device may combine advantageous characteristics of different types of cameras. Moreover, once the relative position of two cameras of the system is calibrated, the problem becomes estimating the motion of an autonomous robot in aid of a binocular head.

## REFERENCES

- [1] M. E. Antone and S. J. Teller. Scalable extrinsic calibration of omnidirectional image networks. In *International Journal of Computer Vision (IJCV)*, vol. 49, pp. 143-174, 2002.
- [2] J. P. Barreto and H. Araujo. Issues on the geometry of central catadioptric image formation. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, vol. 2, pp. 422-427, 2001.
- [3] H. Bay, V. Ferrari and L.J. Van Gool. Wide-baseline stereo matching with line segments. In *CVPR*, pp. 329-336, 2005.
- [4] J. C. Bazin, C. Démonceaux and P. Vasseur. Fast Central Catadioptric Line Extraction. In *3rd Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA'07)*, Lecture Notes in Computer Science, vol. 4478, Pages 25-32, Girone, Espagne, Juin 2007.
- [5] J.C. Bazin, C. Démonceaux, P. Vasseur and I.S. Kweon. Motion estimation by decoupling rotation and translation in catadioptric vision. In *Computer Vision and Image Understanding (CVIU)*, 2009.
- [6] P. David, D. Dementhon, R. Duraiswami and H. Samet. Simultaneous pose and correspondence determination using line features. In *CVPR*, vol. 2, pp. 424, 2003.
- [7] C. Geyer and K. Daniilidis. A unifying theory for central panoramic systems and practical implications. In *Proc. of the European Conference on Computer Vision (ECCV)*, pp. 445-461, 2000.
- [8] P. Gros, O. Bournez and E. Boyer. Using local planar geometric invariants to match and model images of line segments. In *CVIU*, vol. 69, no. 2, pp. 135-155, 1998.
- [9] R.I. Hartley. Defense of the Eight-Point Algorithm. In *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, vol. 19, no. 6, pp. 580-593, June 1997.
- [10] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge University, 2nd edition, 2003.
- [11] A. Heyden. Projective structure and motion from image sequences using subspace methods. In *Proc. 10th Scandinavian Conf. Image Analysis*, pp. 963-968, 1997.
- [12] B. K. P. Horn. Closed-form solution of absolute orientation using unit quaternions. In *Journal of the Optical Society of America. A*, vol. 4, no. 4, pp. 629-642, 1987.
- [13] D. Jacobs. Linear Fitting with missing data: Applications to structure from motion and to characterizing intensity images. In *CVPR*, pp. 206-212, 1997.
- [14] F. Kahl. Multiple view geometry and the  $L_\infty$ -norm. In *Proc. of the 10th IEEE International Conf. on Computer Vision (ICCV)*, pp. II: 1002-1009, 2005.
- [15] F. Kahl and R. Hartley. Multiple-View Geometry Under the  $L_\infty$ -Norm. In *PAMI*, vol. 30, pp. 1603-1617, 2008.
- [16] Q. Ke and T. Kanade. Uncertainty models in quasiconvex optimization for geometric reconstruction. In *CVPR*, pp. 1199-1205, 2006.
- [17] J.H. Kim and R. Hartley. Translation estimation from omnidirectional images. In *Digital Image Computing: Techniques and Applications (DICTA)*, pp. 22, 2005.
- [18] H.C. Longuet-Higgins, A Computer Algorithm for Reconstructing a Scene from Two Projections. In *Nature*, vol. 293, pp. 133-135, 1981.
- [19] S. Mahamud and M. Hebert. Iterative projective reconstruction from multiple views. In *CVPR*, vol. 2, pp. 430-437, 2000.
- [20] A. Makadia, C. Geyer, and K. Daniilidis. Correspondence-free structure from motion. In *IJCV*, vol. 75, pp. 311-327, 2007.
- [21] D. Martinec and T. Pajdla. 3D reconstruction by fitting low-rank matrices with missing data. In *CVPR*, vol. I, pp. 198-205, San Diego, CA, USA, June 2005.
- [22] D. Martinec and T. Pajdla. Robust Rotation and Translation Estimation in Multiview Reconstruction. In *CVPR*, pp. 1-8, 2007.
- [23] C. Mei. Laser-augmented omnidirectional vision for 3D localisation and mapping. PhD Thesis, 2007.
- [24] K. Sim and R. Hartley. Recovering camera motion using the  $L_\infty$ -Norm. In *CVPR*, pp. 1230-1237, 2006.
- [25] P. Sturm and B. Triggs. A factorization based algorithm for multi-image projective structure and motion. In *ECCV*, pp. 709-720, 1996.
- [26] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: a factorization method. In *IJCV(9)*, no. 2, pp. 137-154, November 1992.
- [27] A. Torii, A. Imiya and N. Ohnishi. Two- and three- view geometry for spherical cameras. In *In Proc. IEEE Workshop on Omnidirectional Vision (OMNIVIS05)*.
- [28] P.H.S. Torr and A.W. Fitzgibbon. Invariant fitting of two view geometry. In *PAMI*, vol. 26, no. 5, pp. 648-650, May 2004.
- [29] B. Triggs, P.F. McLauchlan, R.I. Hartley and A.W. Fitzgibbon. Bundle Adjustment - A Modern Synthesis. In *ICCV*, pp. 298-372, 1999.
- [30] X.H. Ying and Z.Y. Hu. Can We Consider Central Catadioptric Cameras and Fisheye Cameras within a Unified Imaging Model. In *ECCV*, vol. I, pp. 442-455, 2004.

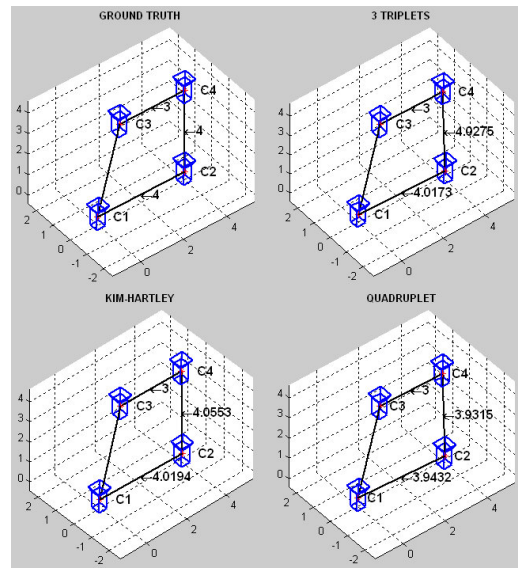


Fig. 4. Recovery of fisheye camera position using 3-triplet, quadruplet and Kim-Hartley [17] methods

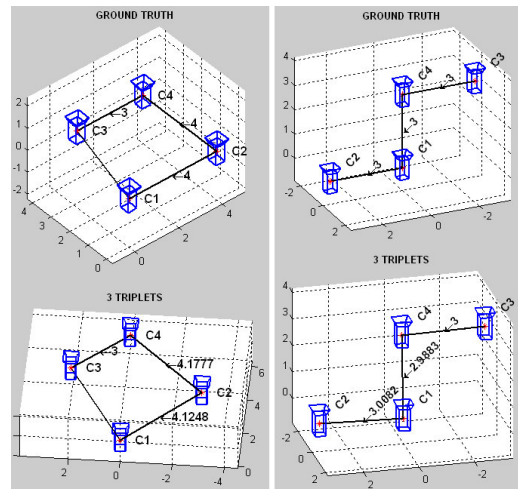


Fig. 5. Position recovery of paracatadioptric camera (left) and perspective camera (right) using 3-triplet approach