Structure and Motion Problems for Multiple Rigidly Moving Cameras

Henrik Stewenius and Kalle Åström

Centre for Mathematical Sciences,
Lund University
{stewe,kalle}@maths.lth.se

Abstract. Vision (both using one-dimensional and two-dimensional retina) is useful for the autonomous navigation of vehicles. In this paper the case of a vehicle equipped with multiple cameras with non-overlapping views is considered. The geometry and algebra of such a moving platform of cameras are considered. In particular we formulate and solve structure and motion problems for a few novel cases of such moving platforms. For the case of two-dimensional retina cameras (ordinary cameras) there are two minimal cases of three points in two platform positions and two points in three platform positions. For the case of one-dimensional retina cameras there are three minimal structure and motion problems. In this paper we consider one of these (6 points in 3 platform positions). The theory has been tested on synthetic data.

1 Introduction

Vision (both using one-dimensional and two-dimensional retina) is useful for the autonomous navigation of vehicles. An interesting case is here when the vehicle is equipped with multiple cameras with different focal points pointing in different directions.

Our personal motivation for this work stems from the **autonomously guided vehicles**, called **AGV**, which are important components for factory automation. Such vehicles have traditionally been guided by wires buried in the factory floor. This gives a very rigid system. The removal and change of wires are cumbersome and costly. The system can be drastically simplified using navigation methods based on laser or vision sensors and computer vision algorithms. With such a system the position of the vehicle can be computed instantly. The vehicle can then be guided along any feasible path in the room.

Note that the discussion here is focused on finding initial estimates of structure and motion. In practice it is necessary to refine these estimates using nonlinear optimisation or bundle adjustment, cf. [Sla80,Åst96].

Structure and motion recovery from a sequence of images is a classical problem within computer vision. A good overview of the techniques available for structure and motion recovery can be found in [HZ00]. Much is known about minimal cases, feature detection, tracking and structure and motion recovery

T. Pajdla and J. Matas (Eds.): ECCV 2004, LNCS 3023, pp. 252–263, 2004.

has been built, [Nis01,PKVG98]. Such systems are however difficult to build. There are ambiguous configurations [KHÅ01] for which structure and motion recovery is impossible. Many automatic systems rely on small image motions in order to solve the correspondence problem. In combination with most cameras small field of view, this limits the way the camera can be moved in order to make good 3D reconstruction. The problem is significantly more stable with a large field of view [OÅ98]. Recently there have been attempts at using rigs with several simple cameras in order to overcome this problem, cf. [Ple03,BFA01]. In this paper we study and solve some of the minimal cases for multi-camera platforms. Such solutions are necessary components for automatic structure and motion recovery systems.

The paper is organised as follows. In section 2 there are some common theory and problem formulation and in the following sections, the studies of three minimal cases.

2 The Geometry of Vision from a Moving Platform

In this paper we consider a platform (a vehicle, robot, car) moving with a planar motion. This vehicle has a number of cameras with different camera centres facing outwards so that they cover different viewing angles. The purpose here is to get a large combined field of view using simple and cheap cameras. The cameras are assumed to have different camera centres but it is assumed that the cameras are calibrated relative to the vehicle, i.e. it is assumed that the camera matrices $\mathbf{P}_i$ are known for all cameras on the vehicle. It is known that a very large field of view improves stability [BFA01] and Pless derives the basic equations to deal with multiple cameras [Ple03].

As both 1D and 2D-retina cameras are studied the equations of both will be introduced in parallel, 1D cameras on the left hand and 2D on the right hand side of the paper.

Both types can be modeled $\lambda \mathbf{u} = \mathbf{P}\mathbf{U}$, where the camera matrix $\mathbf{P}$ is a 2×3 or 3×4 matrix. A scene point $\mathbf{U}$ is in P^2 or P^3 and a measured image point

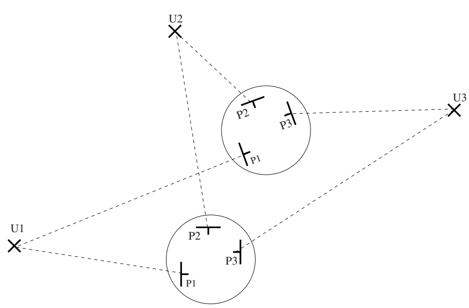

Fig. 1. Three calibrated cameras with constant and known relative positions taking two images each

$\mathbf{u}$ is in P^1 or P^2. It is sometimes useful to consider dual image coordinates. In the one-dimensional retina case each image point $\mathbf{u}$ is dual to a vector $\mathbf{v}$, with $\mathbf{vu} = 0$ and in the 2D case there are two linearly independent dual vectors $\mathbf{v}^1$ and $\mathbf{v}^2$ with $\mathbf{v}^i\mathbf{u} = 0$. The measurement equation then becomes

$$\mathbf{vPU} = 0 \quad \text{or} \quad \mathbf{v}^1\mathbf{PU} = \mathbf{v}^2\mathbf{PU} = 0.$$

As the platform moves the cameras move together. This is modeled as a transformation $\mathbf{S}_i$ between the first position and position i. In the original coordinate system the camera matrix for camera j at position i is $\mathbf{P}_j\mathbf{S}_i$.

It is assumed here that the camera views do not necessarily have common points. In other words, a point is typically seen by only one camera. On the other hand it can be assumed that in a couple of neighboring frames a point can be seen in the same camera. Assume here that point j is visible in camera j. The measurement equation for the n points is then

$$\lambda_{ij}\mathbf{u}_{ij} = \mathbf{P}_j\mathbf{S}_i\mathbf{U}_j, \quad j = 1, \ldots, n, i = 1, \ldots, m.$$

Using the dual image coordinates we obtain

$$\mathbf{v}_{ij}\mathbf{P}_j\mathbf{S}_i\mathbf{U}_j = 0 \quad \text{or} \quad \begin{cases} \mathbf{v}_{ij}^1\mathbf{P}_j\mathbf{S}_i\mathbf{U}_j = 0, \\ \mathbf{v}_{ij}^2\mathbf{P}_j\mathbf{S}_i\mathbf{U}_j = 0, \end{cases} \quad j = 1, \ldots, n, i = 1, \ldots, m,$$

for the one respectively two-dimensional retina case.

Note that $\mathbf{l}_{ij}^T = \mathbf{v}_{ij}\mathbf{P}_j$ and $\mathbf{l}_{ij}^{k\,T} = \mathbf{v}_{ij}^k\mathbf{P}_j$ correspond to the viewing line or viewing plane in the vehicle coordinate system. Thus the constraint can be written

$$\mathbf{l}_{ij}\mathbf{S}_i\mathbf{U}_j = 0 \quad \text{or} \quad \begin{cases} \mathbf{l}_{ij}^1\mathbf{S}_i\mathbf{U}_j = 0, \\ \mathbf{l}_{ij}^2\mathbf{S}_i\mathbf{U}_j = 0, \end{cases} \quad j = 1, \ldots, n, i = 1, \ldots, m. \tag{1}$$

Here the lines or planes $\mathbf{l}$ are measured. The question is if one can calculate structure $\mathbf{U}_j$ and motion $\mathbf{S}_i$ from these measurements. Based on the previous sections, the structure and motion problem will now be defined.

Problem 1. Given the mn images u_{ij} of n points from m different platform positions and the camera matrices P_j the **surveying problem** is to find reconstructed points $\mathbf{U}_j$ and platform transformations $\mathbf{S}_i$ such that

$$\lambda_{ij}\mathbf{u}_{ij} = \mathbf{P}_j\mathbf{S}_i\mathbf{U}_j, \qquad \forall i = 1, \ldots, m, j = 1, \ldots, n$$

for some λ_{ij}.

2.1 Minimal Cases

In order to understand how much information is needed in order to solve the structure and motion problem, it is useful to calculate the number of degrees of freedom of the problem and the number of constraints given by the projection equation. Each object point has two degrees of freedom in the two-dimensional world and three in the three-dimensional world. Vehicle location for a planarily moving vehicle has three degrees of freedom when using $a_i^2 + b_i^2 = 1$, that is Euclidian information and four degrees of freedom in the similarity case when not using this information. The word "image" is used to mean all the information collected by all our cameras at one instant in time.

For Euclidian reconstruction in three dimensions there are $2mn-(3n+3(m-1))$ excess constraints and as seen in table 1 there are two interesting cases

1. two images and three points (m=2,n=3).
2. three images and two points (m=3,n=2).

For the similarity case in two dimensions there are $mn - (2n + 4(m - 1))$ excess constraints and as seen in table 1 there are three interesting cases

1. three images of eight points (m=3, n=8).
2. four images of six points (m=4, n=6).
3. six images of five points (m=6, n=5).

For the Euclidean case in two dimensions there are $mn - (2n + 3(m - 1))$ excess constraints and as seen in table 1 there are three interesting cases

1. three images of six points (m=3, n=6).
2. four images of five points (m=4, n=5) overdetermined.
3. five images of four points (m=5, n=4).

All these will be called the **minimal cases of the structure and motion problem**.

Table 1. The number of excess constraints

2D Similarity

n \ m	1	2	3	4	5	6	7
1	-1	-4	-7	-10	-13	-16	-19
2	-2	-4	-6	-8	-10	-12	-14
3	-3	-4	-5	-6	-7	-8	-9
4	-4	-4	-4	-4	-4	-4	-4
5	-5	-4	-3	-2	-1	**0**	1
6	-6	-4	-2	**0**	2	4	6
7	-7	-4	-1	2	5	8	11
8	-8	-4	**0**	4	8	12	16
9	-9	-4	1	6	11	16	21

2D Euclidian

n \ m	1	2	3	4	5	6	7
1	-1	-3	-5	-7	-9	-11	-13
2	-2	-3	-4	-5	-6	-7	-8
3	-3	-3	-3	-3	-3	-3	-3
4	-4	-3	-2	-1	**0**	1	2
5	-5	-3	-1	**1**	3	5	7
6	-6	-3	**0**	3	6	9	12
7	-7	-3	1	5	9	13	17
8	-8	-3	2	7	12	17	22
9	-9	-3	3	9	15	21	27

3D Euclidian

n \ m	1	2	3	4	5	6	7
1	-1	-2	-3	-4	-5	-6	-7
2	-2	-1	**0**	1	2	3	4
3	-3	**0**	3	6	9	12	15
4	-4	1	6	11	16	21	26
5	-5	2	9	16	23	30	37
6	-6	3	12	21	30	39	48
7	-7	4	15	26	37	48	59
8	-8	5	18	31	44	57	70
9	-9	6	21	36	51	66	81

3 Two-Dimensional Retina, Two Positions, and Three Points

Theorem 1. *For three calibrated cameras that are rigidly fixed with a known transformation relative to each other, each taking an image of a point at two distinct times there generally exist one or three real non-trivial solutions. Pure translation is a degenerate case and can only be computed up to scale.*

Proof. See solution procedure.

Departing from equation (1) for two observations of point j gives

$$\underbrace{\begin{bmatrix} \mathbf{l}_{1j}^1 \mathbf{S}_1 \\ \mathbf{l}_{1j}^2 \mathbf{S}_1 \\ \mathbf{l}_{2j}^1 \mathbf{S}_2 \\ \mathbf{l}_{2j}^2 \mathbf{S}_2 \end{bmatrix}}_{M_j} \mathbf{U}_j = \mathbf{0}.$$

As $\mathbf{U}_j \neq \mathbf{0}$ there is a nonzero solution to the above homogeneous linear system i.e.

$$\det M_j = 0. \tag{2}$$

Note that the constraint above is in essence the same as in [Ple03]. The same constraint can be formulated as $L_1^T F L_2 = 0$, where L_1 and L_2 are plücker coordinate vectors for the space lines and F is a 6×6 matrix representing relative motion of the two platforms.

By a suitable choice of coordinate system it can be assumed that $\mathbf{S}_0 = I$, that is, $a_0 = b_0 = c_0 = d_0 = 0$ and to reduce the number of indices we set $a = a_1$,$b = b_1$,$c = c_1$ and $d = d_1$. The planes defined by $\mathbf{l}_{1j}^1$,$\mathbf{l}_{1j}^2$,$\mathbf{l}_{2j}^1$ and $\mathbf{l}_{2j}^2$ all comes from in camera j and pass through this camera centre in the vehicle coordinate system, that is through a common point, implying

$$\det \begin{bmatrix} \mathbf{l}_{1j}^1 \\ \mathbf{l}_{1j}^2 \\ \mathbf{l}_{2j}^1 \\ \mathbf{l}_{2j}^2 \end{bmatrix} = 0.$$

Computing the determinant in equation (2) gives

$$\alpha_a a + \alpha_{aj} b + \alpha_{cj} c + \alpha_{dj} d + \alpha_{Aj}(da + cb) + \alpha_{Bj}(db - ca) = 0$$

where $\alpha_j = \alpha(\mathbf{l}_{1j}^1, \mathbf{l}_{1j}^2, \mathbf{l}_{2j}^1, \mathbf{l}_{2j}^2)$. The assumption of rigid movement is equivalent to $(1 + a)^2 + b^2 - 1 = 0$. With three points observed, one in each of the three cameras the above gives four polynomials in (a, b, c, d). After homogenisation with t the polynomial equations are

$$\begin{cases} f_1 = \alpha_{a1} at + \alpha_{b1} bt + \alpha_{c1} ct + \alpha_{d1} dt + \alpha_{A1}(ad + bc) + \alpha_{B1}(ac - bd) = 0 \\ f_2 = \alpha_{a2} at + \alpha_{b2} bt + \alpha_{c2} ct + \alpha_{d2} dt + \alpha_{A2}(ad + bc) + \alpha_{B2}(ac - bd) = 0 \\ f_3 = \alpha_{a3} at + \alpha_{b3} bt + \alpha_{c3} ct + \alpha_{d3} dt + \alpha_{A3}(ad + bc) + \alpha_{B3}(ac - bd) = 0 \\ f_4 = a^2 + 2at + b^2 = 0. \end{cases} \tag{3}$$

Lemma 1. *$(a,b,c,d,t) = (0,0,0,0,1)$ and $(a,b,c,d,t) = \lambda(1,\pm i,0,0,0)$ are solutions to equation (3).*

Proof. This follows by inserting the solutions in the equations.

3.1 Solving

Equation (3) is solved by using different combinations of the original equations to construct a polynomial in b which gives roots that can be used to solve for the other variables.
As equation (3) is homogeneous it can be de-homogenised by studying two cases, $a = 0$ and $a = 1$.

$\mathbf{a} = \mathbf{0}$ gives $b = 0$ (by $f_4 = 0$). Now remains

$$\begin{cases} t(\alpha_{c1}c + \alpha_{d1}d) = 0 \\ t(\alpha_{c2}c + \alpha_{d2}d) = 0 \\ t(\alpha_{c3}c + \alpha_{d3}d) = 0, \end{cases}$$

which in general has only the trivial solutions $(c,d,t) = (0,0,t)$ and $(c,d,t) = (c,d,0)$. If the 3 equations are linearly dependent more solutions exist on the form $(c,d,t) = (k_c s, k_d s, t)$. This means that pure translation can only be computed up to scale.

$\mathbf{a} = \mathbf{1}$ gives

$$\begin{cases} f_i = \alpha_{ia}t + \alpha_{ib}bt + (\alpha_{ic} + \alpha_{iB})ct + (\alpha_{id} + \alpha_{iA})dt \\ \quad +\alpha_{iA}(d + bc) + \alpha_{iB}(c - bd) = 0 \qquad\qquad i = 1,2,3 \\ f_4 = 1 + 2t + b^2 = 0. \end{cases}$$

The 24 coefficients of the 17 polynomials

$$f_i,\ f_i b,\ f_i b^2, \quad i = 1,2,3$$
$$f_4 db,\ f_4 d,\ f_4 cb,\ f_4 c,\ f_4 b^3,\ f_4 b^2,\ f_4 b,\ f_4 b$$

are ordered in the lex order $t < d < c < b$ into a 17×24 matrix M with one polynomial per row. As no solutions are lost by multiplying by a polynomial and the original polynomials are in the set, this implies

$$MX = \mathbf{0}, \tag{4}$$

where $X = \begin{bmatrix} X_1^T & X_2^T \end{bmatrix}$ and

$$X_1 = \left[tdb^2, tdb, td, tcb^2, tcb, tc, tb^3, tb^2, tb, t, db^3, db^2, db, d, cb^3, cb^2, cb, c\right]^T$$
$$X_2 = \left[b^5, b^4, b^3, b^2, b, 1\right]^T.$$

Dividing $M = \begin{bmatrix} M_1 & M_2 \end{bmatrix}$ where M_1 is 17×18 and M_2 is 17×6, equation (4) can be written $M_1 X_1 + M_2 X_2 = 0$. Unfortunately these pages are too few to

show the resulting matrices but it is easily proven that $\operatorname{rank} M_1 \le 16$, the easiest way to do this is to use the fact that $\operatorname{rank} M_1 = \operatorname{rank} M_1^T$ and it is easy to find two vectors in the null-space of M_2^T. Therefore there exist v s.t. $vM_1 = \mathbf{0}$ and $0 = \mathbf{0}X = vMX = vM_1X_1 + vM_2X_2 = vM_2X_2$ that is, a fifth order polynomial $p_1(b) = vM_2X_2 = 0$. Knowing that $b = \pm i$ are solutions to this equation, a new polynomial $p_2(b) = p_1(b)/(1+b^2)$ can be calculated and then solved for the remaining roots.
It is then possible to use f_4 to solve for t. Knowing t it is possible to change back to the original non-homogeneous variables (a, b, c, d) and solve for c and d.

3.2 Implementation and Code

The implementation is quite straightforward and the code is available for download [Ste]. As f_4 have fixed coefficients and the result is divided by $(1 + b^2)$ it is possible to reduce M to a 9×12 matrix with M_1 of size 9×10 or 9×8 if the missing rank is used as well. M_2 is of size 9×6 but can be simplified to 9×4 as the roots $b = \pm i$ are known. Reducing the size of M is useful as the highest number of computations come from finding the null-space of M_1.

4 Two-Dimensional Retina, Three Positions, and Two Points

Theorem 2. *Two cameras mounted rigidly with a known transformation with respect to each other for which calibration as well as relative positions are known are moved planarily to 3 different stations where they observe one point per camera. Under these circumstances there generally exist one or three non-trivial real solutions.*
Pure translation is a degenerate case and can only be computed up to scale.

Proof. The existence of a solver giving that number of solutions.

A point j observed in 3 images of the same camera gives

$$\underbrace{\begin{bmatrix} \mathbf{l}_{0j}^1 \mathbf{S}_0 \\ \mathbf{l}_{0j}^2 \mathbf{S}_0 \\ \mathbf{l}_{1j}^1 \mathbf{S}_1 \\ \mathbf{l}_{1j}^2 \mathbf{S}_1 \\ \mathbf{l}_{2j}^1 \mathbf{S}_2 \\ \mathbf{l}_{2j}^2 \mathbf{S}_2 \end{bmatrix}}_{M_j} \mathbf{U}_j = 0.$$

As a non-zero solution exists $\operatorname{rank} M_j \le 3$. This is equivalent to the fact that $\det(M_{sub}) = 0$ for all 4×4 sub-matrices of M. With a suitable choice of coordinate system $\mathbf{S}_0 = I$, that is, $a_0 = b_0 = c_0 = d_0 = 0$ and the 15 sub-matrices of

M_j can be computed and this gives 15 polynomial equations of the second and third degree in $(a_1, b_1, c_1, d_1, a_2, b_2, c_2, d_2)$. Rigid planar motion implies

$$a_i^2 + 2a_i + b_i^2 = 0 \quad i = 1, 2.$$

Observing two points through different but relatively fixed cameras cameras at 3 instants gives total of 32 equations in 8 unknowns.

4.1 Solving

A solver inspired by the previous case has been built but it is still very slow. Anyhow it generates a fifth order polynomial in b_1 for which we know the solutions $b_1 = \pm i$.
If $a_1 = a_2 = 0$ then $b_1 = b_2 = 0$ and the system is degenerate and the solution for (c_1, d_1, c_2, d_2) can only be computed up to scale.

4.2 Implementation and Code

The solver is still in an early stage and very slow. The matrix M will in this case be much larger than in the previous case but the simplifications used there will be possible here as well. As soon as a decent implementation is ready it will be available for download.

5 One-Dimensional Retina, Three Positions, and Six/Eight Points

5.1 Intersection and the Discrete Trilinear Constraint

In this section we will try to use the same technique for solving the structure and motion problem as in [ÅO00]. The idea is to study the equations for a particular point for three views. The fact that three planar lines intersect in a point gives a constraint that we are going to study.

The case of three cameras is of particular importance. Using three measured bearings from three different known locations, the object point is found by intersecting three lines. This is only possible if the three lines actually do intersect. This gives an additional constraint, which can be formulated in the following way

Theorem 3. *Let $\mathbf{l}_{1,j}$, $\mathbf{l}_{2,j}$ and $\mathbf{l}_{3,j}$ be the bearing directions to the same object point from three different camera states. Then the trilinear constraint*

$$\sum_{p,q,r} \mathbf{T}^{pqr} \mathbf{l}_{1j,p} \mathbf{l}_{2j,q} \mathbf{l}_{3j,r} = 0 \,, \tag{5}$$

is fulfilled for some $3 \times 3 \times 3$ tensor $\mathbf{T}$.

Proof. By lining up the line equations

$$\underbrace{\begin{pmatrix} \mathbf{l}_{1,j}\mathbf{S}_1 \\ \mathbf{l}_{2,j}\mathbf{S}_2 \\ \mathbf{l}_{3,j}\mathbf{S}_3 \end{pmatrix}}_{M} \left(\mathbf{U}_j\right) = \mathbf{0}$$

we see that the 3×3 matrix M has a non-trivial right-nullspace. Therefore its determinant is zero. Since the determinant is linear in each row it follows that it can be written as

$$\det M = \sum_{p,q,r} \mathbf{T}^{pqr}\mathbf{l}_{1j,p}\mathbf{l}_{2j,q}\mathbf{l}_{3j,r} = 0\,,$$

for some $3\times3\times3$ tensor $\mathbf{T}$. Here $\mathbf{l}_{1j,p}$ denotes element p of vector $\mathbf{l}_{1j}$ and similarly for $\mathbf{l}_{2j,q}$ and $\mathbf{l}_{3j,r}$.

The tensor $\mathbf{T} = \mathbf{T}^{pqr}$ in (5) will now be analysed in more detail. The mapping from the motion parameters $(\mathbf{S}_1, \mathbf{S}_2, \mathbf{S}_3)$ to the tensor $\mathbf{T}$ is invariant to changes of the coordinate system, i.e. by multiplying each of the transformation matrices with the same matrix. Thus without loss of generality one may assume that $\mathbf{S}_1 = I$. Introducing parameterisations according to (1) the tensor components are

$$\begin{cases} \mathbf{T}^{111} = b_2c_3 - b_3c_2, & \mathbf{T}^{112} = b_2d_3 - a_3c_2, & \mathbf{T}^{113} = b_2, \\ \mathbf{T}^{121} = a_2c_3 - b_3d_2, & \mathbf{T}^{122} = a_2d_3 - a_3d_2, & \mathbf{T}^{123} = a_2, \\ \mathbf{T}^{131} = -b_3, & \mathbf{T}^{132} = -a_3, & \mathbf{T}^{133} = 0, \\ \mathbf{T}^{211} = -a_2c_3 + a_3c_2, & \mathbf{T}^{212} = -a_2d_3 - b_3c_2, & \mathbf{T}^{213} = -a_2, \\ \mathbf{T}^{221} = b_2c_3 + a_3d_2, & \mathbf{T}^{222} = b_2d_3 - b_3d_2, & \mathbf{T}^{223} = b_2, \\ \mathbf{T}^{231} = a_3, & \mathbf{T}^{232} = -b_3, & \mathbf{T}^{233} = 0, \\ \mathbf{T}^{311} = a_2b_3 - a_3b_2, & \mathbf{T}^{312} = a_2a_3 + b_3b_2, & \mathbf{T}^{313} = 0, \\ \mathbf{T}^{321} = -b_3b_2 - a_2a_3, & \mathbf{T}^{322} = a_2b_3 - a_3b_2, & \mathbf{T}^{323} = 0, \\ \mathbf{T}^{331} = 0, & \mathbf{T}^{332} = 0, & \mathbf{T}^{333} = 0. \end{cases} \tag{6}$$

Note that the tensors have a number of zero components. It can in be shown that there are 15 linearly independent linear constraints on the tensor components. These are

$$\begin{cases} \mathbf{T}^{133} = \mathbf{T}^{233} = \mathbf{T}^{313} = \mathbf{T}^{323} = \mathbf{T}^{331} = \mathbf{T}^{332} = \mathbf{T}^{333} = 0, \\ \mathbf{T}^{131} - \mathbf{T}^{232} = \mathbf{T}^{132} + \mathbf{T}^{231} = \mathbf{T}^{113} - \mathbf{T}^{223} = 0 \\ \mathbf{T}^{123} + \mathbf{T}^{213} = \mathbf{T}^{311} - \mathbf{T}^{322} = \mathbf{T}^{321} + \mathbf{T}^{312} = 0, \\ \mathbf{T}^{111} - \mathbf{T}^{122} - \mathbf{T}^{212} - \mathbf{T}^{221} = 0, \\ \mathbf{T}^{112} - \mathbf{T}^{121} - \mathbf{T}^{211} - \mathbf{T}^{222} = 0. \end{cases} \tag{7}$$

There are also four non-linear constraints, i.e.

$$\mathbf{T}^{312}(\mathbf{T}^{123}\mathbf{T}^{131}+\mathbf{T}^{231}\mathbf{T}^{223})+\mathbf{T}^{311}(\mathbf{T}^{123}\mathbf{T}^{231}-\mathbf{T}^{131}\mathbf{T}^{223})=0, \tag{8}$$

$$\begin{aligned}&-\mathbf{T}^{231}\mathbf{T}^{223}\mathbf{T}^{131}\mathbf{T}^{221}+\mathbf{T}^{223}\mathbf{T}^{231}\mathbf{T}^{231}\mathbf{T}^{121}-\mathbf{T}^{123}\mathbf{T}^{231}\mathbf{T}^{231}\mathbf{T}^{111}\\&+\mathbf{T}^{231}\mathbf{T}^{223}\mathbf{T}^{131}\mathbf{T}^{111}-\mathbf{T}^{123}\mathbf{T}^{131}\mathbf{T}^{131}\mathbf{T}^{221}+\mathbf{T}^{123}\mathbf{T}^{131}\mathbf{T}^{231}\mathbf{T}^{121}\\&+\mathbf{T}^{123}\mathbf{T}^{131}\mathbf{T}^{231}\mathbf{T}^{211}-\mathbf{T}^{223}\mathbf{T}^{131}\mathbf{T}^{131}\mathbf{T}^{211}=0,\end{aligned} \tag{9}$$

$$\begin{aligned}&\mathbf{T}^{131}\mathbf{T}^{123}\mathbf{T}^{222}-\mathbf{T}^{131}\mathbf{T}^{121}\mathbf{T}^{123}-\mathbf{T}^{131}\mathbf{T}^{123}\mathbf{T}^{211}+\mathbf{T}^{231}\mathbf{T}^{111}\mathbf{T}^{123}\\&-\mathbf{T}^{131}\mathbf{T}^{122}\mathbf{T}^{223}-\mathbf{T}^{231}\mathbf{T}^{121}\mathbf{T}^{223}=0,\end{aligned} \tag{10}$$

$$\begin{aligned}&\mathbf{T}^{231}\mathbf{T}^{123}\mathbf{T}^{123}\mathbf{T}^{222}-\mathbf{T}^{123}\mathbf{T}^{231}\mathbf{T}^{122}\mathbf{T}^{223}-\mathbf{T}^{123}\mathbf{T}^{231}\mathbf{T}^{221}\mathbf{T}^{223}\\&-\mathbf{T}^{131}\mathbf{T}^{123}\mathbf{T}^{123}\mathbf{T}^{221}-\mathbf{T}^{223}\mathbf{T}^{131}\mathbf{T}^{123}\mathbf{T}^{222}+\mathbf{T}^{131}\mathbf{T}^{122}\mathbf{T}^{223}\mathbf{T}^{223}\\&+\mathbf{T}^{231}\mathbf{T}^{121}\mathbf{T}^{223}\mathbf{T}^{223}+\mathbf{T}^{223}\mathbf{T}^{131}\mathbf{T}^{121}\mathbf{T}^{123}=0.\end{aligned} \tag{11}$$

If only Euclidean transformations of the platform are allowed, which is reasonable, there are two additional constraints

$$(\mathbf{T}^{123})^2+(\mathbf{T}^{113})^2=a_2^2+b_2^2=1, \tag{12}$$

$$(\mathbf{T}^{231})^2+(\mathbf{T}^{232})^2=a_3^2+b_3^2=1. \tag{13}$$

These two last constraints are true only if the tensors are considered to be normalised with respect to scale. It is straightforward to generate corresponding non-normalised (and thus also homogeneous) constraints.

It is natural to think of the tensor as being defined only up to scale. Two tensors $\mathbf{T}$ and $\widetilde{\mathbf{T}}$ are considered equal if they differ only by a scale factor

$$\mathbf{T}\sim\widetilde{\mathbf{T}}.$$

Let $\mathcal{T}$ denote the set of equivalence classes of trilinear tensors fulfilling equations (7)-(13).

Definition 1. *Let the manifold of* **relative motion** *of three platform positions be defined as the set of equivalence classes of three transformations*

$$\mathcal{M}_S=\left\{(\mathbf{S}_1,\mathbf{S}_2,\mathbf{S}_3)\,|\,\mathbf{S}_I=\begin{pmatrix}a_I & b_I & c_I\\ -b_I & a_I & d_I\\ 0 & 0 & 1\end{pmatrix}\right\}\Big/\simeq,$$

where the equivalence is defined as

$$(\mathbf{S}_1,\mathbf{S}_2,\mathbf{S}_3)\simeq(\widetilde{\mathbf{S}}_1,\widetilde{\mathbf{S}}_2,\widetilde{\mathbf{S}}_3),\quad \exists\mathbf{S}\in\mathcal{S},\widetilde{\mathbf{S}}_I\sim\mathbf{S}_I\mathbf{S},I=1,2,3\,.$$

Thus the above discussion states that the map $(\mathbf{S}_1,\mathbf{S}_2,\mathbf{S}_3)\mapsto\mathbf{T}$ is in fact a well defined map from the manifold of equivalence classes $\mathcal{M}_S$ to $\mathcal{T}$.

Theorem 4. *A tensor $\mathbf{T}^{pqr}$ is a calibrated trilinear tensor if and only if equations (7)-(13) are fulfilled. When these constraints are fulfilled it is possible to solve (6) for $\mathbf{S}_2$, $\mathbf{S}_3$. The solution is in general unique.*

Corollary 1. *The map*

$$\mathbf{T}:\mathcal{M}_S\longrightarrow\mathcal{T}$$

is a well defined one-to-one mapping.

5.2 Algorithm

The previous section on the calibrated trilinear tensor has provided us with the tool for solving the structure and motion problem for three platform positions of at least eight points.

Algorithm 51 (Structure and Motion from Three Platform Motions)

1. *Given three images of at least six points,*

$$\mathbf{u}_{ij}, \quad i = 1, \ldots, 3, j = 1, \ldots, n, n \geq 6 \,.$$

2. *Calculate all possible trilinear tensors* $\mathbf{T}$ *that fulfills the constraints (7) to (13) and* $\sum_{p,q,r} \mathbf{T}^{pqr} \mathbf{l}_{1j,p} \mathbf{l}_{2j,q} \mathbf{l}_{3j,r} = 0, \forall j = 1, \ldots n.$
3. *Calculate the platform motions* $(\mathbf{S}_1, \mathbf{S}_2, \mathbf{S}_3)$ *from* $\mathbf{T}$ *according to the proof of Theorem 4.*
4. *For each solution to the motion calculate structure using intersection.*

5.3 Homotopy Studies of Six Points in Three Stations

In step 2 of the above algorithm one has to find all solutions to a system of polynomial equations. We have not yet solved this system in detail, but rather experimented with simulated data and a numerical solver of such system of equations.

When studying the equations of six points and three stations in two dimensions under Euclidian assumption ($a_i^2 + b_i^2 = 1$) in the dual points formulation there are 8 unknowns and 8 variables. By inserting these equations into the homotopy software **PHC-pack** [Ver99] it is found that the mixed volume [CLO98] is 39 and there are up to 25 real solutions.

Based on these experimental investigations we postulate that there are in general up to 25 real solutions to the problem of 6 points in 3 images.

6 Conclusions

In this paper we have introduced the structure and motion problem for the notion of a platform of moving cameras. Three particular cases, (i) eight points in three one-dimensional views, (ii) three points in two two-dimensional views and (iii) two points in three two-dimensional views have been studied. Solutions to these problems are useful for structure and motion estimation of autonomous vehicles equipped with multiple cameras.
The existence of a fast solver for the two images and three points case in three dimensions is of interest when computing RANSAC. It is important to note that pure translation is a degenerate case and that the solution in this case suffers from the same unknown scale as for single camera solutions. Another important aspect is that the cameras has to have separate focal points.

References

[ÅO00] K. Åström and M. Oskarsson. Solutions and ambiguities of the structure and motion problem for 1d retinal vision. *Journal of Mathematical Imaging and Vision*, 12(2):121–135, 2000.

[Åst96] K. Åström. *Invariancy Methods for Points, Curves and Surfaces in Computational Vision*. PhD thesis, Dept of Mathematics, Lund University, Sweden, 1996.

[BFA01] P. Baker, C. Fernmuller, and Y. Aloimonos. A spherical eye from multiple cameras (makes better models of the world). In *Proc. Conf. Computer Vision and Pattern Recognition, Hawaii, USA*, 2001.

[CLO98] D. Cox, J. Little, and D. O'Shea. *Using Algebraic Geometry*. Springer Verlag, 1998.

[GH97] R Gupta and R. I. Hartley. Linear pushbroom cameras. *Pattern Analysis and Machine Intelligence*, 19(9):963–975, sep 1997.

[HZ00] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[KHÅ01] F. Kahl, R. Hartley, and K. Åström. Critical configurations for n-view projective reconstruction. In *Proc. Conf. Computer Vision and Pattern Recognition, Hawaii, USA*, 2001.

[MAS97] C. B. Madsen, C. S. Andersen, and J. S. Sørensen. A robustness analysis of triangulation-based robot self-positioning. In *The 5th Symposium for Intelligent Robotics Systems, Stockholm, Sweden*, 1997.

[Nis01] D. Nistér. *Automatic Dense Reconstruction from Uncalibrated Video Sequences*. PhD thesis, Dept. of Numerical Analysis and Computer Science, Royal Institute of Technology, Stockholm, Sweden, 2001.

[NRT97] J. Neira, I. Ribeiro, and J. D. Tardos. Mobile robot localization and map building using monocular vision. In *The 5th Symposium for Intelligent Robotics Systems, Stockholm, Sweden*, pages 275–284, 1997.

[OÅ98] M. Oskarsson and K. Åström. Accurate and automatic surveying of beacon positions for a laser guided vehicle. In *Proc. European Consortium for Mathematics in Industry, Gothenburg, Sweden*, 1998.

[PKVG98] M. Pollefeys, R. Koch, and L. Van Gool. Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters. In *Proc. 6th Int. Conf. on Computer Vision, Mumbai, India*, 1998.

[Ple03] R. Pless. Using many cameras as one. In *Proc. Conf. Computer Vision and Pattern Recognition, Madison, Canada*, 2003.

[Sla80] C. C. Slama. *Manual of Photogrammetry*. American Society of Photogrammetry, Falls Church, VA, 1980.

[Ste] H. Stewenius. Homepage with code for planar multicamera navigation. http://www.maths.lth.se/˜ stewe/as3/.

[Tri95] B. Triggs. Matching constraints and the joint image. In *Proc. 5th Int. Conf. on Computer Vision, MIT, Boston, MA*, pages 338–343, 1995.

[Ver99] J. Verschelde. Phcpack: A general-purpose solver for polynomial systems by homotopy continuation. *ACM Transactions on Mathematical Software*, 25(2):251–276, 1999.