International Journal of Computer Vision 59(2), 115–137, 2004

Rank Conditions on the Multiple-View Matrix*

YI MA AND KUN HUANG
Electrical & Computer Engineering Department, University of Illinois at Urbana-Champaign, 1406 West Green Street, Urbana, IL 61801, USA
yima@uiuc.edu
kunhuang@uiuc.edu

RENÉ VIDAL
Department of Biomedical Engineering, Johns Hopkins University, Baltimore, MD 21218, USA
rvidal@cis.jhu.edu

JANA KOŠECKÁ
Computer Science Department, George Mason University, 4400 University Dr. #MSA4, Fairfax, VA 22030, USA
kosecka@cs.gmu.edu

SHANKAR SASTRY
Electrical Engineering and Computer Sciences, University of California at Berkeley, Berkeley, CA 94720, USA
sastry@eecs.berkeley.edu

Received October 12, 2001; Revised September 10, 2003; Accepted September 10, 2003

Abstract. Geometric relationships governing multiple images of points and lines and associated algorithms have been studied to a large extent separately in multiple-view geometry. The previous studies led to a characterization based on multilinear constraints, which have been extensively used for structure and motion recovery, feature matching and image transfer. In this paper we present a universal rank condition on the so-called multiple-view matrix M for arbitrarily combined point and line features across multiple views. The condition gives rise to a complete set of constraints among multiple images. All previously known multilinear constraints become simple instantiations of the new condition. In particular, the relationship between bilinear, trilinear and quadrilinear constraints can be clearly revealed from this new approach. The theory enables us to carry out global geometric analysis for multiple images, as well as systematically characterize all degenerate configurations, without breaking image sequence into pairwise or triple-wise sets of views. This global treatment allows us to utilize all incidence conditions governing all features in all images simultaneously for a consistent recovery of motion and structure from multiple views. In particular, a rank-based multiple-view factorization algorithm for motion and structure recovery is derived from the rank condition. Simulation results are presented to validate the multiple-view matrix based approach.

Keywords: multiple-view matrix, rank conditions, structure from motion, factorization

*This work is supported by UIUC ECE Department startup fund, by U.S. Army Research Office under Contract DAAD19-00-1-0466, by the NSF KDI initiative under subcontract SBC-MIT-5710000330, and by the NSF CAREER awards IIS-0347456 and IIS-0347774.

1. Introduction

The basic formulation of the geometric constraints governing projections of point features in two views originated in photogrammetry at the beginning of last century (Kruppa, 1913) and was revived in the computer vision community in the early eighties (Longuet-Higgins, 1981). Natural extensions (of theoretical importance and with profound practical implications) had been those considering multiple views and different feature primitives. In the computer vision literature, fundamental and structure independent relationships between image features and camera displacements were first described by the so-called multilinear matching constraints (Spetsakis and Aloimonos, 1990; Faugeras and Mourrain, 1995; Triggs, 1995; Heyden and Astrom, 1997). Most of the previous work focused on the algebraic aspects of these multilinear constraints, along with the algorithms which followed from the same formulation. This line of work culminated recently with the publication of two monographs on this topic (Hartley and Zisserman, 2000; Faugeras et al., 2001).

Although algebraic and geometric relationships among constraints governing multiple images have been used in numerous applications, theoretical and algorithmic aspects of multiple-view geometry seem to grow in separate directions. In theory, it is well-known (but actually the understanding is not complete as we will shortly see) that relationships among multiple images all depend on those among two, three, or four views at the time. Hence the study of the associated bifocal (Longuet-Higgins, 1981), trifocal (Spetsakis and Aloimonos, 1990; Shashua, 1994; Hartley, 1994) and quadrifocal tensors (Triggs, 1995; Shashua and Wolf, 2000) has been of primary interest in the past few years. However, there is no clear consensus on how to systematically and simultaneously exploit those constraints among pairwise, triple-wise or quadruple-wise images for a consistent 3-D analysis or reconstruction from multiple images. Many existing methods depend on a particular choice of a (sufficient but minimal) set of cascaded pairwise, triple-wise and quadruple-wise constraints (Avidan and Shashua, 1998). Given that such a choice is by no means unique—in fact, the number of choices grows exponentially with the number of images—performance of such methods is very hard to evaluate or justify. Hence, in many practical algorithms "global" and "direct" methods, such as *factorization method* (Tomasi and Kanade, 1992; Triggs, 1996; Kahl and Heyden, 1999; Quan and Kanade, 1996), are sought instead in order to use all data simultaneously. However, such global algorithms rely very little on the theory of multilinear constraints. So, why is there such a separation between theory and algorithms? Is there any way of restating the relationship among multiple images that is equivalent to the multilinear constraints but much easier to use for global analysis and reconstruction? This paper intends to resolve some of that mystery by providing global conditions and constraints on multiple views which facilitate elegant geometric analysis and can be used for development of new algorithms.

There is yet another separation in the current theory of multiple-view geometry itself. To a large extent, different image features: points, lines and planes, and different incidence relations among these features: inclusion, intersection, and restriction, were studied and presented separately, or case by case at best. Hence, it is very difficult to incorporate all features and all incidence relations for a global and consistent reconstruction, using the above mentioned multifocal tensor approach. Besides unnecessary confusion caused by such a separation, answers to many important theoretical questions were left obscured: For instance, since all the constraints are nothing but incidence relations, what is then a universal way of expressing invariants under any transformations which preserve incidence conditions?

The main contribution of our work here is the proof of new and general rank conditions on a formal *multiple-view matrix*, which combines measurements from multiple views of point and line features. Such conditions are proven to be a unified way of capturing *all* types of incidence relations present in multiple-view geometry. This result generalizes recently proposed rank conditions developed separately for point, line, and planar features (Ma et al., 2001), and it certainly completes some previous efforts in the literature that use both line and point features for structure from motion recovery (Liu et al., 1990; Hartley, 1994; Spetsakis and Aloimonos, 1990). The rank conditions on the multiple-view matrix clearly reveal the relationship among all previously known multilinear constraints and further imply some *novel* non-multilinear constraints among multiple images. As a simple byproduct of our approach, we will be able to rigorously show that bilinear and trilinear constraints are sufficient to describe all algebraic relations among multiple *corresponding* views of a point or line feature;

and quadrilinear constraints arise only for the images of a family of intersecting lines. Furthermore, the multiple-view matrix systematically generalizes previously studied trilinear constraints involving mixed point and line features to a multiple-view setting. Hence, it allows for a meaningful global analysis of arbitrarily many views with arbitrarily mixed incidence relations among point, line, and plane features, with no need to cascade pairwise, triple-wise or quadruple-wise views.

The linear structure of the multiple-view rank condition directly facilitates feature matching, feature transfer across multiple views, and motion and structure recovery. The presented formulation also enables a clear geometric characterization and classification of all degenerate configurations in the multiple-view setting, which was intractable using multifocal tensor based methods. As we will see, all degenerate cases will simply correspond to a drop of rank for the multiple-view matrix. An additional appeal of our approach is the sole use of linear algebraic techniques, with no need to introduce tensorial notation or algebraic geometry. Most importantly, the multiple-view rank condition directly leads to a natural class of multiple-view factorization algorithms (for the perspective case) which can incorporate point, line, plane features and any incidence relations among them. The factorization algorithms, which follow from the theory, enable us to recover 3-D structure and camera motion from multiple perspective views of the scene both in case of calibrated and uncalibrated cameras, in a spirit similar to Triggs (1996).

Paper Outline. Section 2 introduces the notation used in this paper as well as the basic concepts and equations for the formulation of the multiple-view matrix. Section 3 gives the proofs for rank conditions on some special multiple-view matrices for point, line and various incidence relations among them. Section 4 summarizes and generalizes all the rank conditions into a single condition on a universal multiple-view matrix, from which *all* multiple-view rank conditions can be instantiated. The geometric interpretation of the rank conditions on the matrix M is given in Section 5. Section 6 outlines some ideas on how to use the multiple-view matrix of mixed features to incorporate all incidence conditions in a scene for a consistent motion and structure recovery. Some simulation results in Section 7 will demonstrate the benefits of the proposed approach.

2. Problem Formulation: Image and Coimage

Let $\mathbb{E}^3$ denote the three-dimensional Euclidean space and $p \in \mathbb{E}^3$ denote a point in the space. The homogeneous coordinates of p relative to a fixed world coordinate frame are denoted as $\boldsymbol{X} \doteq [X, Y, Z, 1]^T \in \mathbb{R}^4$. Then the (perspective) image $\boldsymbol{x}(t) \doteq [x(t), y(t), z(t)]^T \in \mathbb{R}^3$ of p, taken by a moving camera at time t satisfies the following relationship:

$$\lambda(t)\boldsymbol{x}(t) = K(t)Pg(t)\boldsymbol{X}, \tag{1}$$

where $\lambda(t) \in \mathbb{R}_+$ is the (unknown) depth of the point p relative to the camera frame, $K(t) \in SL(3)$ is the camera calibration matrix (at time t),[1] $P = [I, 0] \in \mathbb{R}^{3\times4}$ is the standard (perspective) projection matrix and $g(t) \in SE(3)$ is the coordinate transformation from the world frame to the camera frame at time t.[2] In Eq. (1), $\boldsymbol{x}$, $\boldsymbol{X}$ and g are all in *homogeneous representation*. Suppose the transformation g is specified by its rotation $R \in SO(3)$ and translation $T \in \mathbb{R}^3$,[3] then the homogeneous representation of g is simply:

$$g = \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \in \mathbb{R}^{4\times4}. \tag{2}$$

Notice that Eq. (1) is also equivalent to:

$$\lambda(t)\boldsymbol{x}(t) = [K(t)R(t) \quad K(t)T(t)]\boldsymbol{X}. \tag{3}$$

Now suppose that p is lying on a straight line $L \subset \mathbb{E}^3$, as shown in Fig. 1. The line L can be defined by

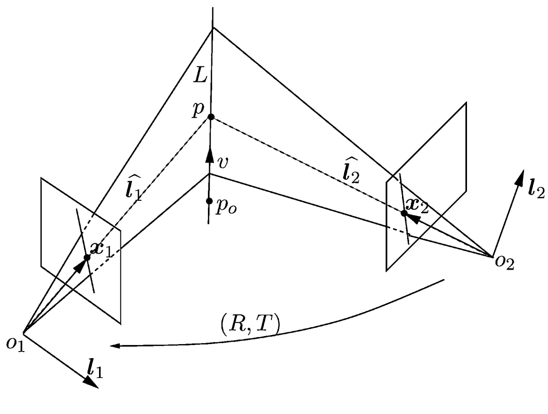

Figure 1. Images of a point p on a line $\boldsymbol{L}$. Planes extended from the image lines $\widehat{\boldsymbol{l}_1}, \widehat{\boldsymbol{l}_2}$ should intersect at the line L in 3-D. Lines extended from the image points $\boldsymbol{x}_1, \boldsymbol{x}_2$ intersect at the point p.

a collection of points in $\mathbb{E}^3$ that can be described (in homogeneous coordinates) as:

$$L \doteq \{\boldsymbol{X} \mid \boldsymbol{X} = \boldsymbol{X}_o + \mu v,\ \mu \in \mathbb{R}\} \subset \mathbb{R}^4, \tag{4}$$

where $\boldsymbol{X}_o = [X_o, Y_o, Z_o, 1]^T \in \mathbb{R}^4$ are coordinates of a base point p_o on this line and $v = [v_1, v_2, v_3, 0]^T \in \mathbb{R}^4$ is a nonzero vector indicating the direction of the line. Then the *image* of the line L at time t is simply the collection of images $\boldsymbol{x}(t)$ of all points $p \in L$. It is clear that all such $\boldsymbol{x}(t)$'s span a plane in $\mathbb{R}^3$, as shown in Fig. 1. The projection of the line is simply the intersection of this plane with the image plane. Usually it is more convenient to specify a plane by its normal vector, denoted as $\boldsymbol{l}(t) = [a(t), b(t), c(t)]^T \in \mathbb{R}^3$. We call this vector $\boldsymbol{l}$ the *coimage* of the line L, which satisfies the following equation:

$$\boldsymbol{l}(t)^T \boldsymbol{x}(t) = \boldsymbol{l}(t)^T K(t) P g(t) \boldsymbol{X} = 0 \tag{5}$$

for any image $\boldsymbol{x}(t)$ of any point p on the line L. Let us use $\hat{u} \in \mathbb{R}^{3\times 3}$ to denote the skew-symmetric matrix associated to a vector $u \in \mathbb{R}^3$ such that $\hat{u}w = u \times w$ for all $w \in \mathbb{R}^3$. Then the column (or row) vectors of the matrix $\hat{\boldsymbol{l}}$ span the image of the line L, i.e., they span the plane which is *orthogonal* to $\boldsymbol{l}$.[4] This is illustrated in Fig. 1. Similarly, if $\boldsymbol{x}$ is the image of a point p, its coimage (the plane orthogonal to $\boldsymbol{x}$) is given by the matrix $\hat{\boldsymbol{x}}$. So in this paper, we will use the following notation:

$$\begin{aligned} &\text{Image of a point: } \boldsymbol{x} \in \mathbb{R}^3, \text{ Coimage of a point: } \hat{\boldsymbol{x}} \in \mathbb{R}^{3\times 3},\\ &\text{Image of a line: } \hat{\boldsymbol{l}} \in \mathbb{R}^{3\times 3}, \text{ Coimage of a line: } \boldsymbol{l} \in \mathbb{R}^3. \end{aligned} \tag{6}$$

Notice that we always have $\hat{\boldsymbol{x}} \cdot \boldsymbol{x} = 0$ and $\hat{\boldsymbol{l}} \cdot \boldsymbol{l} = 0$. Since image and coimage are equivalent representation of the same geometric entity, sometimes for simplicity (and by abuse of language) we may simply refer to either one as "image" if its meaning is clear from the context.

In a realistic situation, we usually obtain "sampled" images of $\boldsymbol{x}(t)$ or $\boldsymbol{l}(t)$ at some time instances: $t_1, t_2, \ldots, t_m$. For simplicity we denote:

$$\lambda_i = \lambda(t_i),\ \ \boldsymbol{x}_i = \boldsymbol{x}(t_i),\ \ \boldsymbol{l}_i = \boldsymbol{l}(t_i),\ \ \Pi_i = K(t_i) P g(t_i). \tag{7}$$

We will call the matrix Π_i the *projection matrix* relative to the ith camera frame. The matrix Π_i is then a 3×4 matrix which relates the ith image of the point p to its world coordinates $\boldsymbol{X}$ by:

$$\lambda_i \boldsymbol{x}_i = \Pi_i \boldsymbol{X} \tag{8}$$

and the ith coimage of the line L to its world coordinates $(\boldsymbol{X}_o, v)$ by:

$$\boldsymbol{l}_i^T \Pi_i \boldsymbol{X}_o = \boldsymbol{l}_i^T \Pi_i v = 0, \tag{9}$$

for $i = 1, 2, \ldots, m$. If the point is actually lying on the line, we further have a relationship between the image of the point and the coimage of the line:

$$\boldsymbol{l}_i^T \boldsymbol{x}_i = 0, \tag{10}$$

for $i = 1, 2, \ldots, m$.

We first observe that the unknowns, λ_i's, $\boldsymbol{X}$, $\boldsymbol{X}_o$ and v, which encode the information about location of the point p or the line L in $\mathbb{R}^3$ are not intrinsically available from the images. By eliminating these unknowns from the equations we obtain the remaining intrinsic relationships between $\boldsymbol{x}, \boldsymbol{l}$ and Π only, i.e. between the image measurements and the camera configuration. Of course there are many different, but algebraically equivalent ways for elimination of these unknowns. This has in fact resulted in different kinds (or forms) of multilinear (or multifocal) constraints that exist in the computer vision literature. We here introduce a more *systematic* way of eliminating *all* the above unknowns that results in a *complete* set of conditions and a clear geometric characterization of *all* constraints. Consequently, as we will soon see, all previously known and even some unknown relationships can be easily deduced from our results.

3. Special Multiple-View Matrices and Their Ranks

After rewriting Eq. (8) in matrix form, we observe that for the point p, the basic relationship between its image measurements and camera motions after eliminating the unknowns λ_i's and $\boldsymbol{X}$ is, that the $m + 4$ column vectors of the following matrix:

$$N_p \doteq \begin{bmatrix} \Pi_1 & \boldsymbol{x}_1 & 0 & \cdots & 0 \\ \Pi_2 & 0 & \boldsymbol{x}_2 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ \Pi_m & 0 & \cdots & 0 & \boldsymbol{x}_m \end{bmatrix} \in \mathbb{R}^{3m\times(m+4)} \tag{11}$$

are *linearly dependent*. Equivalently:

$$\text{rank}(N_p) \leq m + 3 \tag{12}$$

since it is direct to verify that the vector $u \doteq [\boldsymbol{X}^T, -\lambda_1, \ldots, -\lambda_m]^T \in \mathbb{R}^{m+4}$ is in the null space of the matrix N_p due to (8). On the other hand, given the matrix N_p, if its rank is exactly $m + 3$, we may find the depths of the point (relative to all camera frames) by solving the equation $N_p u = 0$. However, if the rank of N_p is strictly less than $m + 3$, there might be more solutions. Hence we need to study in more detail the possible values of the rank of N_p and understand their geometric implications. From here, our approach will deviate from previous ones which directly take minors of the matrix N_p and derive multifocal constraints. Our observation here is that the matrix N_p is rather sparse and well-structured. It is then natural to try to make it more compact before computing any minors. As we will soon see, one may gain tremendous insight in this way.

Similarly, one can show that for line features the following matrix:

$$N_l \doteq \begin{bmatrix} \boldsymbol{l}_1^T \Pi_1 \\ \boldsymbol{l}_2^T \Pi_2 \\ \vdots \\ \boldsymbol{l}_m^T \Pi_m \end{bmatrix} \in \mathbb{R}^{m \times 4} \tag{13}$$

must satisfy the rank condition

$$\text{rank}(N_l) \leq 2 \tag{14}$$

since it is clear that the vectors $\boldsymbol{X}_o$ and v are both in the null space of the matrix N_l due to (9). In fact, any $\boldsymbol{X} \in \mathbb{R}^4$ in the null space of N_l represents the homogeneous coordinates of some point lying on the line L, and vice versa.

The above rank conditions on N_p and N_l are merely the starting point. There is some potential difficulty if one wants to use them directly since: 1. the lower bounds on their rank are not yet clear; 2. their dimensions are high and hence the rank conditions contain a lot of redundancy. Let us see how to systematically reduce these conditions to more compact ones.

3.1. Rank Condition for Point Features

Let us first study the point case. Without loss of generality, we may assume that the first camera frame is chosen to be the reference frame.[5] That gives the projection matrices $\Pi_i, i = 1, 2, \ldots, m$

$$\Pi_1 = [I, 0], \ldots, \Pi_i = [R_i, T_i], \ldots, \Pi_m = [R_m, T_m] \in \mathbb{R}^{3\times 4}, \tag{15}$$

where $R_i \in \mathbb{R}^{3\times 3}$ represents the first three columns of Π_i and $T_i \in \mathbb{R}^3$ is the fourth column of $\Pi_i, i = 2, 3, \ldots, m$. Although we have used the suggestive notation (R_i, T_i) here, they are not necessarily the actual rotation and translation.[6] Only in the case when the camera is perfectly calibrated do R_i and T_i correspond to actual camera rotation and translation. The internal structure of R_i and T_i is not so important for the rest of the paper, as long as R_i is invertible.

With the above notation, we eliminate $\boldsymbol{x}_1$ from the first row of N_p using column manipulation. It is easy to see that N_p has the same rank as the following matrix in $\mathbb{R}^{3m\times(m+4)}$:

$$\begin{bmatrix} I & 0 & 0 & 0 & \cdots & 0 \\ R_2 & T_2 & R_2\boldsymbol{x}_1 & \boldsymbol{x}_2 & \ddots & \vdots \\ \vdots & \vdots & \vdots & 0 & \ddots & 0 \\ R_m & T_m & R_m\boldsymbol{x}_1 & 0 & 0 & \boldsymbol{x}_m \end{bmatrix} = \left[\begin{array}{c|c} I & 0 \\ \hline R_2 & \\ \vdots & N_p' \\ R_m & \end{array}\right].$$

Hence, the original N_p is rank deficient if and only if the sub-matrix $N_p' \in \mathbb{R}^{3(m-1)\times(m+1)}$ is. Left multiplying N_p' by the following matrix:

$$D_p = \begin{bmatrix} \boldsymbol{x}_2^T & 0 & \cdots & 0 \\ \widehat{\boldsymbol{x}_2} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \boldsymbol{x}_m^T \\ 0 & \cdots & 0 & \widehat{\boldsymbol{x}_m} \end{bmatrix} \in \mathbb{R}^{[4(m-1)]\times[3(m-1)]}$$

yields the following matrix:

$$D_p N_p' = \begin{bmatrix} \boldsymbol{x}_2^T T_2 & \boldsymbol{x}_2^T R_2 \boldsymbol{x}_1 & \boldsymbol{x}_2^T \boldsymbol{x}_2 & 0 & 0 & 0 \\ \widehat{\boldsymbol{x}_2} T_2 & \widehat{\boldsymbol{x}_2} R_2 \boldsymbol{x}_1 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & 0 & \ddots & 0 & 0 \\ \vdots & \vdots & 0 & 0 & \ddots & 0 \\ \boldsymbol{x}_m^T T_m & \boldsymbol{x}_m^T R_m \boldsymbol{x}_1 & 0 & 0 & 0 & \boldsymbol{x}_m^T \boldsymbol{x}_m \\ \widehat{\boldsymbol{x}_m} T_m & \widehat{\boldsymbol{x}_m} R_m \boldsymbol{x}_1 & 0 & 0 & 0 & 0 \end{bmatrix} \in \mathbb{R}^{[4(m-1)]\times(m+1)}.$$

Since D_p is of full rank $3(m-1)$, we have $\text{rank}(N'_p) = \text{rank}(D_p N'_p)$. Further, since $\boldsymbol{x}_i^T \boldsymbol{x}_i > 0$, the original matrix N_p is rank deficient if and only if the following sub-matrix of $D_p N'_p$ is rank deficient:

$$M_p \doteq \begin{bmatrix} \widehat{\boldsymbol{x}_2} R_2 \boldsymbol{x}_1 & \widehat{\boldsymbol{x}_2} T_2 \\ \widehat{\boldsymbol{x}_3} R_3 \boldsymbol{x}_1 & \widehat{\boldsymbol{x}_3} T_3 \\ \vdots & \vdots \\ \widehat{\boldsymbol{x}_m} R_m \boldsymbol{x}_1 & \widehat{\boldsymbol{x}_m} T_m \end{bmatrix} \in \mathbb{R}^{[3(m-1)]\times 2}. \quad (16)$$

We call M_p the *multiple-view matrix* associated to a point feature p. More precisely, we have proven the following:

Lemma 1 (*Rank condition for point features*). *Matrices N_p and M_p satisfy*:

$$0 \leq \text{rank}(M_p) = \text{rank}(N_p) - (m+2) \leq 1. \quad (17)$$

Therefore rank(N_p) is either $m+3$ or $m+2$, depending on whether rank(M_p) is 1 or 0, respectively.

Comment 1 (*Geometric interpretation of M_p*). *Notice that $\widehat{\boldsymbol{x}_i} T_i$ is the normal to the epipolar plane given by frames 1 and i and so is $\widehat{\boldsymbol{x}_i} R_i \boldsymbol{x}_1$. Therefore the rank-1 condition not only implies that these two normals are parallel (as obvious from the epipolar constraint) but also that the scale between these two possible normal vectors is the same for all frame pairs. In fact, if rank(M_p) $= 1$, its kernel must be $[\lambda_1, 1]^T \in \mathbb{R}^2$. Hence the depth of the point p (relative to the reference frame) is encoded in M_p too. See Ma et al. (2001) for more details.*

Corollary 1 (*Algebraic relationships among multiple views of a point*). *For any given m images of a point $p \in \mathbb{E}^3$ relative to m camera frames, the rank condition on M_p implies the following*:

1. *Any algebraic constraints among m images algebraically depend on those involving only two and three images at a time. Formulae of these constraints are given by* (18) *and* (19) *respectively.*
2. *The three-view constraints* (19) *in general imply the two-view constraints* (18), *except when $\widehat{\boldsymbol{x}_i} T_i = \widehat{\boldsymbol{x}_i} R_i \boldsymbol{x}_1 = 0$ for some i. This corresponds to a degenerate case in which the point p lies on the line through the optical centers o_1, o_i.*

Proof: For the first claim, notice that the rank reduction process is equivalent to (real) algebraic manipulations of minors of the corresponding matrices M_p and N_p. The reduction process only eliminates nonzero factors from the minors. Hence the algebraic constraints generated from the minors of M_p are algebraically equivalent to those obtained from N_p (in the ring of polynomials over the real field).

Now, the matrix M_p in (16) is rank deficient if and only if all its 2×2 minors have determinant zero. Since all such minors involve images $\boldsymbol{x}_i$ from up to three views, these constraints are sufficient to characterize all relationships among the m views. For the two columns of M_p to be linearly dependent, it is necessary that the two vectors $\widehat{\boldsymbol{x}_i} T_i, \widehat{\boldsymbol{x}_i} R_i \boldsymbol{x}_1$ of its ith row group are linearly dependent. This gives the well-known *bilinear constraints* between the ith and 1st images:

$$\boldsymbol{x}_i^T \hat{T}_i R_i \boldsymbol{x}_1 = 0. \quad (18)$$

Now further consider the ith and jth row groups of the matrix M_p together, we obtain:

$$\begin{aligned} (\widehat{\boldsymbol{x}_i} R_i \boldsymbol{x}_1)(\widehat{\boldsymbol{x}_j} T_j)^T - (\widehat{\boldsymbol{x}_i} T_i)(\widehat{\boldsymbol{x}_j} R_j \boldsymbol{x}_1)^T = 0 \\ \Rightarrow \widehat{\boldsymbol{x}_i}\big(T_i \boldsymbol{x}_1^T R_j^T - R_i \boldsymbol{x}_1 T_j^T\big)\widehat{\boldsymbol{x}_j} = 0, \end{aligned} \quad (19)$$

for $i, j = 2, 3, \ldots, m$. Note that for this is a matrix equation, it gives a total of $3 \times 3 = 9$ scalar equations in terms of $\boldsymbol{x}_1, \boldsymbol{x}_i, \boldsymbol{x}_j$. For $i \neq j$, the 9 equations are exactly the *trilinear constraints* that one would obtain from the minors of N_p following the conventional derivation of trilinear constraints.

Hence the bilinear and trilinear constraints are the generators of the ideal of polynomial equations that multiple *corresponding* images of a point need to satisfy.[7] The so-called *quadrilinear constraints* (Triggs, 1995; Shashua and Wolf, 2000), in the point feature case, do not impose any algebraically independent constraints on the four images other than the trilinear and bilinear ones.[8]

For the second claim, notice that, for $i = j$, the trilinear Eq. (19) implies that $\widehat{\boldsymbol{x}_i} T_i$ and $\widehat{\boldsymbol{x}_i} R_i \boldsymbol{x}_1$ are linearly dependent, except the case that both vectors are zeros. One can easily verify from epipolar geometry that this occurs only when the point p lies on the line through the optical centers o_1, o_i (i.e., its images coincide with the epipoles). □

The two possible values for the rank of M_p classify geometric configurations of the point relative to the m

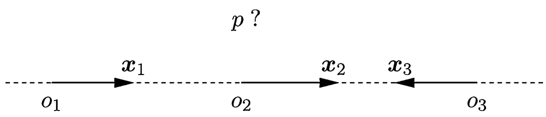

Figure 2. If the three images and the three optical centers lie on a straight line, any point on this line is a valid preimage that satisfies all the constraints.

camera frames into two categories only. If $\text{rank}(M_p) = 1$, then $\text{rank}(N_p) = m + 3$ and there is a unique solution to $N_p u = 0$. If $\text{rank}(M_p) = 0$, however, there are always global degenerate geometric configurations. More explicitly, we have:

Corollary 2 (*Degenerate configurations for point features*). *Given m images corresponding to a point $p \in \mathbb{R}^3$ relative to m camera frames, they correspond to images of a unique point in the 3-D space if the rank of the M_p matrix (relative to any of the camera frames) is 1. If its rank is 0, i.e. $M_p = 0$, the point p is determined up to a line where all the camera centers must lie on, as shown in Fig. 2.*

3.2. Rank Condition for Line Features

As before, we choose the first camera frame as the reference. The matrix N_l then becomes:

$$N_l = \begin{bmatrix} \boldsymbol{l}_1^T & 0 \\ \boldsymbol{l}_2^T R_2 & \boldsymbol{l}_2^T T_2 \\ \vdots & \vdots \\ \boldsymbol{l}_m^T R_m & \boldsymbol{l}_m^T T_m \end{bmatrix} \in \mathbb{R}^{m\times 4}. \tag{20}$$

This matrix should have a rank of no more than 2. Multiplying N_l on the right by the following matrix:

$$D_l = \begin{bmatrix} \widehat{\boldsymbol{l}_1} & \boldsymbol{l}_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{4\times 5} \tag{21}$$

yields:

$$N_l' = \begin{bmatrix} 0 & \boldsymbol{l}_1^T \boldsymbol{l}_1 & 0 \\ \boldsymbol{l}_2^T R_2 \widehat{\boldsymbol{l}_1} & \boldsymbol{l}_2^T R_2 \boldsymbol{l}_1 & \boldsymbol{l}_2^T T_2 \\ \vdots & \vdots & \vdots \\ \boldsymbol{l}_m^T R_m \widehat{\boldsymbol{l}_1} & \boldsymbol{l}_m^T R_m \boldsymbol{l}_1 & \boldsymbol{l}_m^T T_m \end{bmatrix} \in \mathbb{R}^{m\times 5}. \tag{22}$$

Since D_l is of full rank 4, we obtain:

$$\text{rank}(N_l') = \text{rank}(N_l) \leq 2.$$

Since $\boldsymbol{l}_1^T \boldsymbol{l}_1 > 0$, this is true if and only if the following sub-matrix of N_l':

$$M_l \doteq \begin{bmatrix} \boldsymbol{l}_2^T R_2 \widehat{\boldsymbol{l}_1} & \boldsymbol{l}_2^T T_2 \\ \boldsymbol{l}_3^T R_3 \widehat{\boldsymbol{l}_1} & \boldsymbol{l}_3^T T_3 \\ \vdots & \vdots \\ \boldsymbol{l}_m^T R_m \widehat{\boldsymbol{l}_1} & \boldsymbol{l}_m^T T_m \end{bmatrix} \in \mathbb{R}^{(m-1)\times 4} \tag{23}$$

has rank no more than one. We call the matrix M_l the *multiple-view matrix* associated to a line feature L. We have just proven the following:

Lemma 2 (*Rank condition for line features*). *For the two matrices N_l and M_l, we have*:

$$0 \leq rank(M_l) = rank(N_l) - 1 \leq 1. \tag{24}$$

Therefore $rank(N_l)$ is either 2 or 1, depending on whether $rank(M_l)$ is 1 or 0, respectively.

Comment 2 (*Geometric interpretation of M_l*). *Since $rank(M_l) \leq 1$, each row of M_l represents the same (homogeneous) coordinates in $\mathbb{R}^4$. If we normalize them to be $[v_1, v_2, v_3, 1]^T \in \mathbb{R}^4$, the vector $v = [v_1, v_2, v_3, 0]^T \in \mathbb{R}^4$ is exactly the direction of the line L and $1/\|v\|$ is exactly the distance of the line to the camera center of the reference camera frame. See Ma et al. (2001) for more details.*

Corollary 3 (*Relationships among multiple views of a line*). *For any given m images of a line L in $\mathbb{E}^3$ relative to m camera frames, the rank deficient matrix M_l implies that any algebraic constraints among the m images algebraically depend those involving only 3 images at a time, characterized by the so-called trilinear constraints.*

Proof: The rank condition stipulates that constraints among the given m images of the line all reduces to those involving three views at a time. To see this more explicitly, notice that for $\text{rank}(M_l) \leq 1$, it is necessary for any pair of row vectors of M_l to be linearly dependent. From the linear dependency of the two vectors $\boldsymbol{l}_i^T R_i \widehat{\boldsymbol{l}_1}, \boldsymbol{l}_j^T R_j \widehat{\boldsymbol{l}_1}$ from first three columns, we get

$$\boldsymbol{l}_i^T R_i \widehat{\boldsymbol{l}_1} R_j^T \boldsymbol{l}_j = 0. \tag{25}$$

Taking into account the last column, we have the more conventional trilinear constraints for lines:

$$\boldsymbol{l}_j^T T_j \boldsymbol{l}_i^T R_i \widehat{\boldsymbol{l}_1} - \boldsymbol{l}_i^T T_i \boldsymbol{l}_j^T R_j \widehat{\boldsymbol{l}_1} = 0. \quad (26)$$

Both equations are trilinear among the 1st, ith and jth images. Hence the constraint rank$(M_l) \leq 1$ is a natural generalization of the trilinear constraint (for 3 views) to arbitrary m views since when $m = 3$, it is equivalent to the trilinear constraint for lines, except for some degenerate cases that we will discuss later.

It is easy to see from the rank of matrix M_l that there will be no other relationships among either pairwise or quadruple-wise images of a line. Hence trilinear constraints are the only constraints that corresponding images of a single line in 3-D space satisfy.[9] □

Comment 3 (*Constraints on rotations*). *Notice that Eq.* (25) *imposes constraints on the camera rotations R_i only. This type of constraints has been utilized in the literature for reconstruction using line segments, e.g. see Taylor and Kriegman* (1995). *However, rank$(M_l) \leq 1$ obviously gives a more general presentation for all the constraints.*

Although the trilinear constraints (26) are necessary for the rank of matrix M_l (hence N_l) to be 1, rigorously speaking they are *not* sufficient. For the Eq. (26) to be nontrivial, it is required that the entry $\boldsymbol{l}_i^T T_i$ in the involved rows of M_l be nonzero. This is not always true for certain degenerate cases—such as the line being parallel to the translational direction. The rank condition on M_l is certainly a better way of capturing *all* constraints among multiple images and avoids artificial degeneracy that could be introduced by using algebraic equations.

As with M_p, the two possible values for the rank of M_l classify geometric configurations of the line relative to the m camera frames into two categories only. If rank$(M_l) = 1$, rank$(N_l) = 2$ and the coordinates of points on L are uniquely determined by the equation $N_l \boldsymbol{X} = 0$. If rank$(M_l) = 0$, however, there are always global degenerate geometric configurations. More explicitly, we have:

Corollary 4 (*Degenerate configurations for line features*). *Given m vectors in $\mathbb{R}^3$ representing coimages of a line with respect to m camera frames, they correspond to a unique line in the 3-D space if the rank of the matrix M_l relative to any of the camera frames is* 1. *If its rank is* 0, *i.e.*, $M_l = 0$, *then the line is determined up to a plane on which all the camera centers must lie, as shown in Fig.* 3.

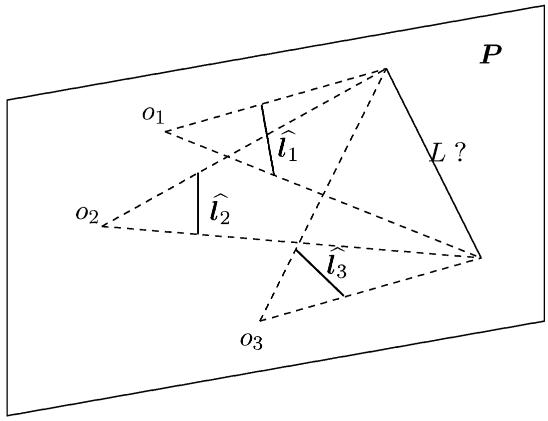

Figure 3. If the three images and the three optical centers lie on a plane *bmP*, any line on this plane would be a valid preimage of $\widehat{l_1}, \widehat{l_2}, \widehat{l_3}$ that satisfies all the constraints.

3.3. Rank Conditions for Arbitrary Incidence Relations

As we have seen in the previous sections, multiple *corresponding* images of a point or a line in $\mathbb{E}^3$ are governed by certain rank conditions. Such conditions not only concisely capture geometric constraints among multiple images, but also are the key to further reconstruction of the camera motion and scene structure. In this section, we will show that all basic incidence relationships among different features, i.e., *inclusion*, *intersection*, or *restriction* of features, can also be fully captured by such rank conditions. Since these relationships can be easily detected or verified in each image, such knowledge can be and should be exploited if a consistent reconstruction is sought.

3.3.1. Inclusion of Features. Consider the situation when you observe a line $\boldsymbol{l}_1$ in the reference view but in remaining views, you observe images $\boldsymbol{x}_2, \ldots, \boldsymbol{x}_m$ of a feature point on the line—we say that this feature point is *included* by the line. To derive the constraints that such features have to satisfy, we start with the matrix N_p and left multiply it by the following matrix:

$$D_l' = \begin{bmatrix} \boldsymbol{l}_1^T & 0 \\ \widehat{\boldsymbol{l}_1} & 0 \\ 0 & I_{3(m-1)\times 3(m-1)} \end{bmatrix} \in \mathbb{R}^{(3m+1)\times 3m}. \quad (27)$$

We obtain:

$$D_l' N_p = \begin{bmatrix} \boldsymbol{l}_1^T & 0 & 0 & 0 & \cdots & 0 \\ \widehat{\boldsymbol{l}_1} & 0 & \widehat{\boldsymbol{l}_1}\boldsymbol{x}_1 & 0 & \cdots & 0 \\ R_2 & T_2 & 0 & \boldsymbol{x}_2 & \ddots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & 0 \\ R_m & T_m & 0 & \cdots & 0 & \boldsymbol{x}_m \end{bmatrix} \in \mathbb{R}^{(3m+1)\times(m+4)}. \tag{28}$$

Since $\text{rank}(D_l') = 3m$, we have $\text{rank}(N_p) = \text{rank}(D_l' N_p) \leq m + 3$. Now left multiply $D_l' N_p$ by the the following matrix:

$$D_p' = \begin{bmatrix} I_{4\times4} & 0 & 0 & \cdots & 0 \\ 0 & \widehat{\boldsymbol{x}_2} & 0 & \cdots & 0 \\ 0 & \boldsymbol{x}_2^T & 0 & \cdots & 0 \\ 0 & 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & 0 & \widehat{\boldsymbol{x}_m} \\ 0 & \cdots & 0 & 0 & \boldsymbol{x}_m^T \end{bmatrix} \in \mathbb{R}^{(4m+1)\times(3m+1)}. \tag{29}$$

It is direct to verify that the rank of the resulting matrix $D_p' D_l' N_p$ is related to the rank of its sub-matrix:

$$N_p'' = \begin{bmatrix} \boldsymbol{l}_1^T & 0 \\ \widehat{\boldsymbol{x}_2} R_2 & \widehat{\boldsymbol{x}_2} T_2 \\ \vdots & \vdots \\ \widehat{\boldsymbol{x}_m} R_m & \widehat{\boldsymbol{x}_m} T_m \end{bmatrix} \in \mathbb{R}^{(3m-2)\times4} \tag{30}$$

by the expression $\text{rank}(N_p'') + m \leq \text{rank}(D_p' D_l' N_p) = \text{rank}(N_p)$. Now right multiplying N_p'' by:

$$\begin{bmatrix} \boldsymbol{l}_1 & \widehat{\boldsymbol{l}_1} & 0 \\ 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{4\times5} \tag{31}$$

yields:

$$\begin{bmatrix} \boldsymbol{l}_1^T \boldsymbol{l}_1 & 0 & 0 \\ \widehat{\boldsymbol{x}_2} R_2 \boldsymbol{l}_1 & \widehat{\boldsymbol{x}_2} R_2 \widehat{\boldsymbol{l}_1} & \widehat{\boldsymbol{x}_2} T_2 \\ \vdots & \vdots & \vdots \\ \widehat{\boldsymbol{x}_m} R_m \boldsymbol{l}_1 & \widehat{\boldsymbol{x}_m} R_m \widehat{\boldsymbol{l}_1} & \widehat{\boldsymbol{x}_m} T_m \end{bmatrix} \in \mathbb{R}^{(3m-2)\times5}. \tag{32}$$

We call its sub-matrix:

$$M_{lp} \doteq \begin{bmatrix} \widehat{\boldsymbol{x}_2} R_2 \widehat{\boldsymbol{l}_1} & \widehat{\boldsymbol{x}_2} T_2 \\ \vdots & \vdots \\ \widehat{\boldsymbol{x}_m} R_m \widehat{\boldsymbol{l}_1} & \widehat{\boldsymbol{x}_m} T_m \end{bmatrix} \in \mathbb{R}^{[3(m-1)]\times4} \tag{33}$$

the *multiple-view matrix* for a point included by a line. Its rank is related to that of N_p by the expression:

$$\text{rank}(M_{lp}) \leq \text{rank}(N_p) - (m+1) \leq 2. \tag{34}$$

Since $\text{rank}(AB) \geq (\text{rank}(A) + \text{rank}(B) - n)$ for all $A \in \mathbb{R}^{m\times n}$, $B \in \mathbb{R}^{n\times k}$, we have $\text{rank}(\widehat{\boldsymbol{x}_i} R_i \widehat{\boldsymbol{l}_1}) \geq 1$. So we essentially have proven the following:

Lemma 3 (*Rank condition for inclusion*). *For multiple images of a point p on a line L, the multiple-view matrix M_{lp} defined above satisfies*

$$1 \leq \mathit{rank}(M_{lp}) \leq \mathit{rank}(N_p) - (m+1) \leq 2. \tag{35}$$

The rank condition on M_{lp} then captures the incidence condition in which a line with coimage $\boldsymbol{l}_1$ includes a point p in $\mathbb{E}^3$ with images $\boldsymbol{x}_2, \ldots, \boldsymbol{x}_m$ with respect to $m-1$ camera frames. What kind of equations does this rank condition give rise to? Without loss of generality, let us look at the sub-matrix

$$M_{lp} = \begin{bmatrix} \widehat{\boldsymbol{x}_2} R_2 \widehat{\boldsymbol{l}_1} & \widehat{\boldsymbol{x}_2} T_2 \\ \widehat{\boldsymbol{x}_3} R_3 \widehat{\boldsymbol{l}_1} & \widehat{\boldsymbol{x}_3} T_3 \end{bmatrix} \in \mathbb{R}^{6\times4}. \tag{36}$$

The rank condition implies that every 3×3 sub-matrix of M_{lp} has determinant zero. The first three rows are automatically of rank 2 and so are the last three rows and the first three columns.[10] Hence any nontrivial determinant consists of two and only two rows from either the first three or last three rows, and consists of two and only two columns from the first three columns. For example, if we choose two rows from the first three, it is direct to see that such a determinant is a polynomial of degree 5 on (the entries of) $\boldsymbol{l}_1, \boldsymbol{x}_2$ and $\boldsymbol{x}_3$ which is quadratic in both $\boldsymbol{l}_1$ and $\boldsymbol{x}_2$. The resulting equation is *not* multilinear in these vectors at all,[11] but it indeed imposes nontrivial constraints among these images (or coimages).

If we have four images (in M_{lp}), one may take one row from the 2nd, 3rd and 4th images. The resulting equation would involve all four images. Notice that

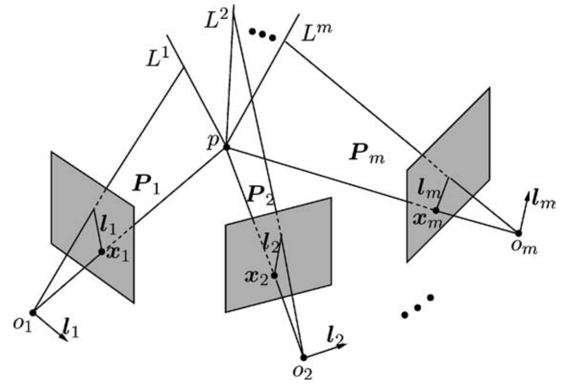

Figure 4. Images of a family of lines $L^1, L^2, \ldots, L^m$ intersect at a point p. Planes $\boldsymbol{P}_1, \boldsymbol{P}_2, \ldots, \boldsymbol{P}_m$ extended from the image lines might not intersect at the same line in 3-D. But all the planes must intersect at the same point.

each row of the matrix $\hat{\boldsymbol{x}}$ in M_{lp} represents (the coimage of) a line $\boldsymbol{l}$ passing through the point $\boldsymbol{x}$. Hence this type of constraint imposes the incidence relations that the associated lines in space all pass through the same point, as shown in Fig. 4, which is exactly equivalent to the conventional quadrilinear constraints (Triggs, 1995).

3.3.2. Intersection of Features. Consider the situation in which, in the reference, you observe the image $\boldsymbol{x}_1$ of a point p, but in the remaining views, you observe coimages $\boldsymbol{l}_2, \boldsymbol{l}_3, \ldots, \boldsymbol{l}_m$ of lines that *intersect* at the point, as shown in Fig. 4. In this case these coimages do not have to correspond to the same line in $\mathbb{E}^3$. There are many different, but equivalent ways to derive the constraints that such set of features has to satisfy (see Ma et al., 2001). For simplicity, we start with the matrix M_p:

$$M_p = \begin{bmatrix} \widehat{\boldsymbol{x}_2} R_2 \boldsymbol{x}_1 & \widehat{\boldsymbol{x}_2} T_2 \\ \widehat{\boldsymbol{x}_3} R_3 \boldsymbol{x}_1 & \widehat{\boldsymbol{x}_3} T_3 \\ \vdots & \vdots \\ \widehat{\boldsymbol{x}_m} R_m \boldsymbol{x}_1 & \widehat{\boldsymbol{x}_m} T_m \end{bmatrix} \in \mathbb{R}^{[3(m-1)]\times 2}. \quad (37)$$

This matrix should have a rank of no more than 1. Since the point belongs to all the lines, we have $\boldsymbol{l}_i^T \boldsymbol{x}_i = 0, i = 1, 2, \ldots, m$. Hence $\boldsymbol{l}_i \in \text{range}(\widehat{\boldsymbol{x}_i})$. That is, there exist $u_i \in \mathbb{R}^3$ such that $\boldsymbol{l}_i = \widehat{\boldsymbol{x}_i}^T u_i, i = 1, 2, \ldots, m$. Since $\text{rank}(M_p) \leq 1$, so should be the rank of following matrix (whose rows are simply linear combinations of those of M_p):

$$\begin{bmatrix} u_2^T \widehat{\boldsymbol{x}_2} R_2 \boldsymbol{x}_1 & u_2^T \widehat{\boldsymbol{x}_2} T_2 \\ \vdots & \vdots \\ u_m^T \widehat{\boldsymbol{x}_m} R_m \boldsymbol{x}_1 & u_m^T \widehat{\boldsymbol{x}_m} T_m \end{bmatrix} = \begin{bmatrix} \boldsymbol{l}_2^T R_2 \boldsymbol{x}_1 & \boldsymbol{l}_2^T T_2 \\ \vdots & \vdots \\ \boldsymbol{l}_m^T R_m \boldsymbol{x}_1 & \boldsymbol{l}_m^T T_m \end{bmatrix} \in \mathbb{R}^{(m-1)\times 2}. \quad (38)$$

We call the matrix

$$M_{pl} \doteq \begin{bmatrix} \boldsymbol{l}_2^T R_2 \boldsymbol{x}_1 & \boldsymbol{l}_2^T T_2 \\ \vdots & \vdots \\ \boldsymbol{l}_m^T R_m \boldsymbol{x}_1 & \boldsymbol{l}_m^T T_m \end{bmatrix} \in \mathbb{R}^{(m-1)\times 2} \quad (39)$$

the *multiple-view matrix* for lines intersecting at a point. Then we have:

Lemma 4 (*Rank condition for intersection*). *Given the image of a point p and multiple images of lines intersecting at p, the multiple-view matrix M_{pl} defined above satisfies*:

$$0 \leq \mathit{rank}(M_{pl}) \leq 1. \quad (40)$$

The above rank condition on the matrix M_{pl} captures the incidence condition between a point and lines which intersect at the same point. It is worth noting that for the rank condition to be true, it is necessary that all 2×2 minors of M_{pl} be zero, i.e., the following constraints hold among arbitrary triplets of given images:

$$\left[\boldsymbol{l}_i^T R_i \boldsymbol{x}_1\right]\left[\boldsymbol{l}_j^T T_j\right] - \left[\boldsymbol{l}_i^T T_i\right]\left[\boldsymbol{l}_j^T R_j \boldsymbol{x}_1\right] = 0 \in \mathbb{R}, \quad i, j = 2, \ldots, m. \quad (41)$$

These are exactly the well-known *point-line-line* relationships among three views (Hartley and Zisserman, 2000). However, here $\boldsymbol{l}_i$ and $\boldsymbol{l}_j$ do not have to be coimages of the same line in $\mathbb{E}^3$. Their preimages only have to intersect at the same point p. This undoubtedly relaxes the restriction on the meaning of "corresponding" line features. Hence our results have extended the use of the point-line-line relationship.

3.3.3. Features Restricted to a Plane. Another incidence condition commonly encountered in practice is that all features involved are actually lying on a plane, say $\boldsymbol{P}$, in $\mathbb{E}^3$. In general, a plane can be described by a vector $\pi = [a, b, c, d] \in \mathbb{R}^4$ such that the homogeneous coordinates $\boldsymbol{X}$ of any point p on this plane satisfy the equation:

$$\pi \boldsymbol{X} = 0. \tag{42}$$

Although we assume such a constraint on the coordinates $\boldsymbol{X}$ of p, in general we do not have to assume that we know π in advance.

Similarly, consider a line $L = \{\boldsymbol{X} \mid \boldsymbol{X} = \boldsymbol{X}_o + \mu v, \mu \in \mathbb{R}\}$. Then this line is in the plane $\boldsymbol{P}$ if and only if:

$$\pi \boldsymbol{X}_o = \pi v = 0. \tag{43}$$

For convenience, given a plane $\pi = [a, b, c, d]$, we usually define $\pi^1 = [a, b, c] \in \mathbb{R}^3$ and $\pi^2 = d \in \mathbb{R}$.

It turns out that, in order to take into account the planar restriction, we only need to change the definition of each multiple-view matrix slightly, but all the rank conditions remain exactly the same. This is because in order to combine Eqs. (8) and (9) with the planar constraints (42) and (43), we only need to change the definition of the matrices N_p and N_l to:

$$N_p \doteq \begin{bmatrix} \Pi_1 & \boldsymbol{x}_1 & 0 & \cdots & 0 \\ \Pi_2 & 0 & \boldsymbol{x}_2 & \ddots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ \Pi_m & 0 & \cdots & 0 & \boldsymbol{x}_m \\ \pi & 0 & \cdots & \cdots & 0 \end{bmatrix} \quad \text{and}$$

$$N_l \doteq \begin{bmatrix} \boldsymbol{l}_1^T \Pi_1 \\ \boldsymbol{l}_2^T \Pi_2 \\ \vdots \\ \boldsymbol{l}_m^T \Pi_m \\ \pi \end{bmatrix}.$$

Such modifications do not change the rank of N_p or N_l. Therefore, as before, we have $\text{rank}(N_p) \leq m + 3$ and $\text{rank}(N_l) \leq 2$. Then one can easily follow the previous proofs for all the rank conditions by carrying this extra row of (planar) constraint with the matrices and the rank conditions on the resulting multiple-view matrices remain the same as before. We summarize the results as follows:

Corollary 5 (*Rank conditions for planar features*). *Given a point p and a line L lying on a plane $\boldsymbol{P}$ which is specified by the vector $\pi \in \mathbb{R}^4$, in Lemmas* 1 *and* 4, *append the matrix $[\pi^1 \boldsymbol{x}_1 \ \pi^2]$ to the matrices M_p and M_{pl}; in Lemmas* 2 *and* 3, *append the matrix $[\pi^1 \widehat{\boldsymbol{l}_1} \ \pi^2]$ to the matrices M_l and M_{lp}. Then the rank conditions on the new matrices M_p, M_l, M_{lp} and M_{pl} remain the same as in Lemmas* 1–4.

For example, the multiple-view matrices M_p and M_l become:

$$M_p = \begin{bmatrix} \widehat{\boldsymbol{x}_2} R_2 \boldsymbol{x}_1 & \widehat{\boldsymbol{x}_2} T_2 \\ \widehat{\boldsymbol{x}_3} R_3 \boldsymbol{x}_1 & \widehat{\boldsymbol{x}_3} T_3 \\ \vdots & \vdots \\ \widehat{\boldsymbol{x}_m} R_m \boldsymbol{x}_1 & \widehat{\boldsymbol{x}_m} T_m \\ \pi^1 \boldsymbol{x}_1 & \pi^2 \end{bmatrix}, \quad M_l = \begin{bmatrix} \boldsymbol{l}_2^T R_2 \widehat{\boldsymbol{l}_1} & \boldsymbol{l}_2^T T_2 \\ \boldsymbol{l}_3^T R_3 \widehat{\boldsymbol{l}_1} & \boldsymbol{l}_3^T T_3 \\ \vdots & \vdots \\ \boldsymbol{l}_m^T R_m \widehat{\boldsymbol{l}_1} & \boldsymbol{l}_m^T T_m \\ \pi^1 \widehat{\boldsymbol{l}_1} & \pi^2 \end{bmatrix}. \tag{44}$$

Then the rank condition $\text{rank}(M_p) \leq 1$ implies not only the multilinear constraints as before, but also the following equations (by considering the sub-matrix consisting of the ith group of three rows of M_p and its last row)

$$\widehat{\boldsymbol{x}_i} T_i \pi^1 \boldsymbol{x}_1 - \widehat{\boldsymbol{x}_i} R_i \boldsymbol{x}_1 \pi^2 = 0, \quad i = 2, \ldots, m. \tag{45}$$

When the plane $\boldsymbol{P}$ does not cross the camera center o_1, i.e., $\pi^2 \neq 0$, these equations give exactly the well-known *homography* constraints for planar image feature points

$$\widehat{\boldsymbol{x}_i} \left(R_i - \frac{1}{\pi^2} T_i \pi^1 \right) \boldsymbol{x}_1 = 0 \tag{46}$$

between the 1st and the ith views. The matrix $H_i = (R_i - \frac{1}{\pi^2} T_i \pi^1)$ in the equation is the well-known *homography matrix* between the two views. Similarly from the rank condition on M_l, we can obtain homography in terms of line features

$$\boldsymbol{l}_i^T \left(R_i - \frac{1}{\pi^2} T_i \pi^1 \right) \widehat{\boldsymbol{l}_1} = 0, \quad i = 2, \ldots, m \tag{47}$$

We know that on a plane $\boldsymbol{P}$, any two points determine a line and any two lines determine a point. This dual relationship is inherited in the following relationship between the rank conditions on M_p and M_l:

Corollary 6 (*Duality between coplanar points and lines*). *If the matrices M_p of two distinct points on a plane are of rank less than or equal to* 1, *then the*

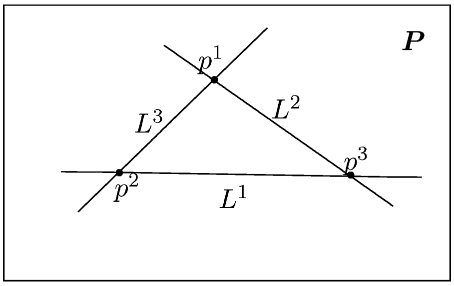

Figure 5. Duality between a set of three points and three lines in a plane $\boldsymbol{P}$: the rank conditions associated to p^1, p^2, p^3 are exactly equivalent those associated to L^1, L^2, L^3.

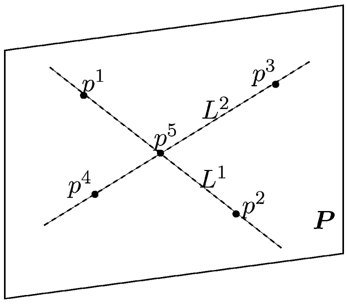

Figure 6. p^1, p^2, p^3, p^4 are four points on the same plane $\boldsymbol{P}$, if and only if the two associated (virtual) lines L^1 and L^2 intersect at a (virtual) point p^5.

matrix M_l associated to the line determined by the two points is of rank less than or equal to 1. *On the other hand, if the matrices M_l of two distinct lines on a plane are of rank less than or equal to* 1, *then the matrix M_p associated to the intersection of the two lines is of rank less than or equal to* 1.

The proof is omitted for simplicity (for details, see Ma et al., 2001). An immediate implication of this corollary is that given a set of feature points sharing the same 3-D plane, it really does not matter too much whether one uses the matrices M_p for points, or the matrices M_l for lines determined by pairwise points (in all the views). They essentially give exactly the same set of constraints. This is illustrated in Fig. 5.

The above approach for expressing a planar restriction relies explicitly on the parameters π of the underlying plane $\boldsymbol{P}$ (which leads to the homography). There is however another intrinsic (but equivalent) way to express the planar restriction, by using combinations of the rank conditions that we have so far on point and line features. Since three points are always coplanar, at least four points are needed to make any planar restriction nontrivial. Suppose four feature points p^1, p^2, p^3, p^4 are coplanar as shown in Fig. 6, and their images are denoted as $\boldsymbol{x}^1, \boldsymbol{x}^2, \boldsymbol{x}^3, \boldsymbol{x}^4$. The (virtual) coimages $\boldsymbol{l}^1, \boldsymbol{l}^2$ of the two lines L^1, L^2 and the (virtual) image $\boldsymbol{x}^5$ of their intersection p^5 can be uniquely determined by $\boldsymbol{x}^1, \boldsymbol{x}^2, \boldsymbol{x}^3, \boldsymbol{x}^4$:

$$\boldsymbol{l}^1 = \widehat{\boldsymbol{x}^1}\boldsymbol{x}^2, \quad \boldsymbol{l}^2 = \widehat{\boldsymbol{x}^3}\boldsymbol{x}^4, \quad \boldsymbol{x}^5 = \widehat{\boldsymbol{l}^1}\boldsymbol{l}^2. \tag{48}$$

Then the coplanar constraint for p^1, p^2, p^3, p^4 can be expressed in terms of the intersection relation between L^1, L^2 and p^5. If we use $\boldsymbol{l}_i^j$ to denote the ith image of the line j, for $j = 1, 2$, and $i = 1, 2, \ldots, m$, and $\boldsymbol{x}_i^j$ is defined similarly. According to the preceding Section 3.3.2, the coplanar constraint is equivalent to the following matrix:

$$\tilde{M}_{pl} = \begin{bmatrix} \boldsymbol{l}_2^{1T} R_2 \boldsymbol{x}_1^5 & \boldsymbol{l}_2^{1T} T_2 \\ \boldsymbol{l}_2^{2T} R_2 \boldsymbol{x}_1^5 & \boldsymbol{l}_2^{2T} T_2 \\ \vdots & \vdots \\ \boldsymbol{l}_m^{1T} R_m \boldsymbol{x}_1^5 & \boldsymbol{l}_m^{1T} T_m \\ \boldsymbol{l}_m^{2T} R_m \boldsymbol{x}_1^5 & \boldsymbol{l}_m^{2T} T_m \end{bmatrix} \in \mathbb{R}^{[2(m-1)]\times 2} \tag{49}$$

satisfying rank($\tilde{M}_{pl}$) ≤ 1 condition. Note that, unlike (45), this condition does not explicitly depends on the parameters π. But it is essentially equivalent to the homography Eq. (45) (with π eliminated using the images of all four coplanar points). Algebraic equations that one may get from this rank condition, again, will not be multilinear in the given $\boldsymbol{x}_i^j$, $j = 1, 2, 3, 4$, $i = 1, 2, \ldots, m$. Instead, these equations are typically *quadratic* in each $\boldsymbol{x}_i^j$, $j = 1, 2, 3, 4$, $i = 1, 2, \ldots, m$. Despite that, we here see again the effectiveness of using rank conditions for *intrinsically* expressing incidence relations.

4. Rank Condition on the Universal Multiple-View Matrix

In preceding sections, we have seen that incidence conditions among multiple images of multiple features can usually be concisely expressed in terms of certain rank conditions on various types of the so-called multiple-view matrix. In this section, we will demonstrate that all these conditions are simply special instantiations of a unified rank condition on a universal multiple-view matrix.

For m images $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m$ of a point p on a line L with its m coimages $\boldsymbol{l}_1, \boldsymbol{l}_2, \ldots, \boldsymbol{l}_m$, we define the following set of matrices:

$$D_i \doteq \boldsymbol{x}_i \in \mathbb{R}^3 \quad \text{or} \quad \widehat{\boldsymbol{l}_i} \in \mathbb{R}^{3\times 3},$$
$$D_i^{\perp} \doteq \widehat{\boldsymbol{x}_i} \in \mathbb{R}^{3\times 3} \quad \text{or} \quad \boldsymbol{l}_i^T \in \mathbb{R}^3.$$

Then, depending on whether the available (or chosen) measurement from the ith image is the point feature $\boldsymbol{x}_i$ or the line feature $\boldsymbol{l}_i$, the D_i (or $D_i^{\perp}$) matrix is assigned its corresponding value. That choice is completely independent of the other D_j (or $D_j^{\perp}$) for $j \neq i$. The "dual" matrix $D_i^{\perp}$ can be viewed as the *orthogonal supplement* to D_i and it always represents a coimage (of a point or a line).[12] Using the above definition of D_i and $D_i^{\perp}$, we now formally define a *universal multiple-view matrix*:

$$M \doteq \begin{bmatrix} D_2^{\perp} R_2 D_1 & D_2^{\perp} T_2 \\ D_3^{\perp} R_3 D_1 & D_3^{\perp} T_3 \\ \vdots & \vdots \\ D_m^{\perp} R_m D_1 & D_m^{\perp} T_m \end{bmatrix}. \qquad (50)$$

Depending on the particular choice for each $D_i^{\perp}$ or D_1, the dimension of the matrix M may vary. But no matter what the choice for each individual $D_i^{\perp}$ or D_1 is, M will always be a valid matrix of certain dimension. Then after elimination of the unknowns λ_i's, $\boldsymbol{X}$, $\boldsymbol{X}_o$ and v in the system of equations in (8) and (9), we obtain:

Theorem 1 (*Multiple-view rank conditions*). *Consider a point p lying on a line L and its images $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_m \in \mathbb{R}^3$ and coimages $\boldsymbol{l}_1, \boldsymbol{l}_2, \ldots, \boldsymbol{l}_m \in \mathbb{R}^3$ relative to m camera frames whose relative configuration is given by (R_i, T_i) for $i = 2, 3, \ldots, m$. Then for any choice of $D_i^{\perp}$ and D_1 in the definition of the multiple-view matrix M, the rank of the resulting M belongs to and only to one of the following two cases*:

1. *If $D_1 = \widehat{\boldsymbol{l}_1}$ and $D_i^{\perp} = \widehat{\boldsymbol{x}_i}$ for some $i \geq 2$, then*:

$$\boxed{1 \leq rank(M) \leq 2.} \qquad (51)$$

2. *Otherwise*:

$$\boxed{0 \leq rank(M) \leq 1.} \qquad (52)$$

A complete proof of this theorem is a straight-forward combination and extension of Lemmas 1–4. Essentially, the above theorem gives a universal description of the incidence relation between a point and line in terms of their m images seen from m vantage points.

As a result of Theorem 1, all previously known and some additional unknown constraints among multiple images of point or line features are simply certain *instantiations* of the rank conditions of Theorem 1. The instantiations corresponding to case 2 are exactly the ones that give rise to the multilinear constraints in the literature. The instantiations corresponding to case 1, as we have seen before, give rise to constraints that are *not* necessarily multilinear. The completeness of Theorem 1 also implies that there would be *no* multilinear relationship among quadruple-wise views, even in the mixed feature scenario.[13] Therefore, quadrilinear constraints and quadrilinear tensors are clearly *redundant* for multiple-view analysis. However, as we mentioned before (in Section 3.3.1), nontrivial constraints may still exist up to four views.

As we have demonstrated in the previous sections, other incidence conditions such as all features belonging to a plane in $\mathbb{E}^3$ can also be expressed in terms of the same set of rank conditions:

Corollary 7 (*Planar features and homography*). *Suppose that all features are in a plane and coordinates $\boldsymbol{X}$ of any point on it satisfy the equation $\pi \boldsymbol{X} = 0$ for some vector $\pi^T \in \mathbb{R}^4$. Denote $\pi = [\pi^1, \pi^2]$ with $\pi^{1^T} \in \mathbb{R}^3, \pi^2 \in \mathbb{R}$. Then simply append the matrix*

$$\left[\pi^1 D_1 \; \pi^2\right] \qquad (53)$$

to the matrix M in its formal definition (50). *The rank condition on the new M remains exactly the same as in Theorem* 1.

The rank condition on the new matrix M then implies *all* constraints among multiple images of these planar features, as well as the special constraint previously known as homography. Of course, the above representation is not intrinsic – it depends on parameters π that describe the 3-D location of the plane. Following the process in Section 3.3.3, the above corollary reduces to rank conditions on matrices of the type in (49), which in turn, give multi-quadratic constraints on the images involved.

Remark 1 (Features at infinity). In Theorem 1, if the point p and the line L are in the plane at infinity $\mathbb{P}^3 \backslash \mathbb{R}^3$, the rank condition on the multiple-view matrix M is just *the same*. Hence the rank condition extends to multiple-view geometry of the entire projective space $\mathbb{P}^3$, and it does not discriminate between Euclidean, affine or projective space. In fact, the rank conditions are invariant under a much larger group of transformations: It allows any transformation that preserves all the incidence relations among a given set of features, these transformations *do not* even have to be linear. For instance, one can fold a piece of paper along its diagonal, which is not a rigid body transformation. Nevertheless, rank conditions associated to images of its vertices and edges remain the same.

Remark 2 (Occlusion). If any feature is occluded in a particular view, the corresponding row (or a group of rows) is simply omitted from M; or if only the point is occluded but not the entire line(s) on which the point lies, then simply replace the missing image of the point by the corresponding image(s) of the line(s). In either case, the overall rank condition on M remains *unaffected*. In fact, the rank condition on M gives a very effective criterion to tell whether or not a set of (mixed) features indeed corresponds one to another. If the features are miss-matched, either due to occlusion or to errors while establishing correspondences, the rank condition will be violated.

5. Geometric Interpretation of the Multiple-View Matrix

Since there are practically infinitely many instantiations for the multiple-view matrix, it is impossible to provide a geometric description for each one of them. Instead, we will discuss a few essential cases that will give the reader a clear idea about how the rank condition of the multiple-view matrix works geometrically. In particular, we will demonstrate that a further drop of rank in the multiple-view matrix M, can be clearly interpreted in terms of a corresponding geometric degeneracy. Understanding these representative cases would be sufficient for the reader to carry out a similar analysis to any other instantiations.

5.1. Case 2: $0 \leq Rank(M) \leq 1$

Let us first consider the more general case, i.e., case 2 in Theorem 1, when $\text{rank}(M) \leq 1$. We will discuss case 1 afterwards. For case 2, there are only two interesting

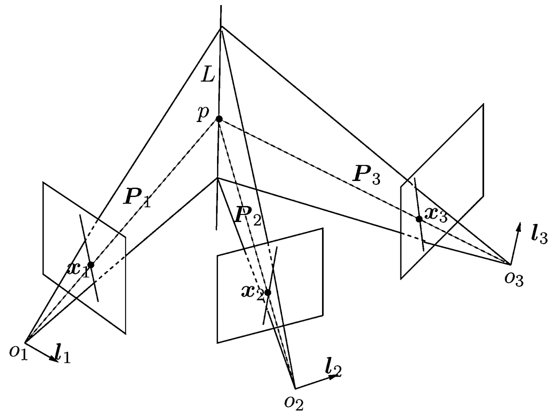

Figure 7. Generic configuration for the case $\text{rank}(M) = 1$. Planes extended from the (co-)images $\boldsymbol{l}_1, \boldsymbol{l}_2, \boldsymbol{l}_3$ intersect at one line L in 3-D. Lines extended from the images $\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3$ intersect at one point p. p must lie on L.

sub-cases, depending on the value of the rank of M, are:

$$(a)\ \text{rank}(M) = 1, \quad \text{and} \quad (b)\ \text{rank}(M) = 0. \quad (54)$$

Case (a), when rank of M is 1, corresponds to the generic case for which, regardless of the particular choice of features in M, all these features satisfy the incidence condition. More explicitly, all the point features (if at least 2 are present in M) come from a unique 3-D point p, all the lines features (if at least 3 are present in M) come from a unique 3-D line L, and if both point and line features are present, the point p then must lie on the line L in 3-D. This is illustrated in Fig. 7.

What happens if there are not enough point or line features present in M? If, for example, there is only one point feature $\boldsymbol{x}_1$ present in M_{pl}, then the rank of M_{pl} being 1 implies that the line L is uniquely determined by $\boldsymbol{l}_2, \ldots, \boldsymbol{l}_m$. Hence, the point p is determined by L and $\boldsymbol{x}_1$. On the other hand, if there is only one line feature present in some M, but more than two point features, L is then a family of lines lying on a plane and passing through the point p determined by the rest of point features in M.

Case (b), when the rank of M is 0, implies that all the entries of M are zero. It is easy to verify that this corresponds to a set of degenerate cases in which the 3-D location of the point or the line cannot be uniquely determined from their multiple images (no matter how many), and the incidence condition between the point

p and the line L no longer holds. In these cases, the best we can do is:

- When there are more than two point features present in M, the 3-D location of the point p can be determined up to a line which connects all camera centers (related to these point features);
- When there are more than three line features present in M, the 3-D location of the line L can be determined up to the plane on which all related camera centers must lie;
- When both point and line features are present in M, we can usually determine the point p up to a line (connecting all camera centers related to the point features) which lies on the same plane on which the rest of the camera centers (related to the line features) and the line L must lie.

Let us demonstrate this last case on a concrete example. Suppose the number of views is $m = 6$ and we choose the matrix M to be:

$$M = \begin{bmatrix} \boldsymbol{l}_2^T R_2 \boldsymbol{x}_1 & \boldsymbol{l}_2^T T_2 \\ \boldsymbol{l}_3^T R_3 \boldsymbol{x}_1 & \boldsymbol{l}_3^T T_3 \\ \boldsymbol{l}_4^T R_4 \boldsymbol{x}_1 & \boldsymbol{l}_4^T T_4 \\ \widehat{\boldsymbol{x}_5} R_5 \boldsymbol{x}_1 & \widehat{\boldsymbol{x}_5} T_5 \\ \widehat{\boldsymbol{x}_6} R_6 \boldsymbol{x}_1 & \widehat{\boldsymbol{x}_6} T_6 \end{bmatrix} \in \mathbb{R}^{9\times 2}. \quad (55)$$

The geometric configuration of the point and line features corresponding to the condition $\text{rank}(M) = 0$ is illustrated in Fig. 8. But notice that, among all the possible solutions for L and p, if they both happen to be at infinity, the incidence condition then would hold for all the images involved.

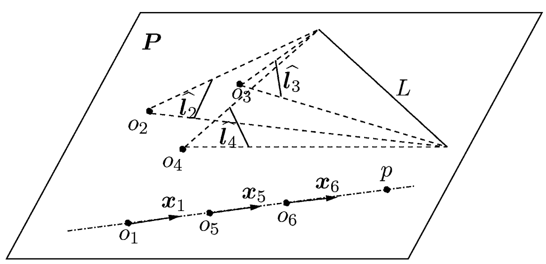

Figure 8. A degenerate configuration for the case $\text{rank}(M) = 0$: a point-line-line-line-point-point scenario. From the given rank condition, the line L could be any where on the plane spanned by all the camera centers; the point p could be any where on the line through o_1, o_5, o_6.

5.2. *Case 1: 1 ≤ Rank(M) ≤ 2*

We now discuss case 1 in Theorem 1, when $\text{rank}(M) \leq 2$. In this case, the matrix M must contain at least one sub-matrix of the type:

$$[\widehat{\boldsymbol{x}_i} R_i \widehat{\boldsymbol{l}_1} \quad \widehat{\boldsymbol{x}_i} T_i] \in \mathbb{R}^{3\times 4}, \quad (56)$$

for some $i \geq 2$. It is easy to verify that such a sub-matrix can never be zero, hence the only possible values for the rank of M are:

$$(a)\ \text{rank}(M) = 2, \quad \text{and} \quad (b)\ \text{rank}(M) = 1. \quad (57)$$

Case (a), when the rank of M is 2, corresponds to the generic cases for which the incidence condition among the features is effective. The essential example here is the matrix M_{lp} given in (33):

$$M_{lp} = \begin{bmatrix} \widehat{\boldsymbol{x}_2} R_2 \widehat{\boldsymbol{l}_1} & \widehat{\boldsymbol{x}_2} T_2 \\ \vdots & \vdots \\ \widehat{\boldsymbol{x}_m} R_m \widehat{\boldsymbol{l}_1} & \widehat{\boldsymbol{x}_m} T_m \end{bmatrix} \in \mathbb{R}^{[3(m-1)]\times 4}. \quad (58)$$

If $\text{rank}(M_{lp}) = 2$, it can be shown that the point p is only determined up to the plane specified by o_1 and $\boldsymbol{l}_1$ but all the point features $\boldsymbol{x}_2, \ldots, \boldsymbol{x}_m$ correspond to the same point p. The line L is only determined up to this plane, but the point p does not have to be on this line. This is illustrated in Fig. 9. Beyond M_{lp}, if there are more than two line features present in some M, the point p then must lie on every plane associated to every line feature. Hence p must be on the intersection

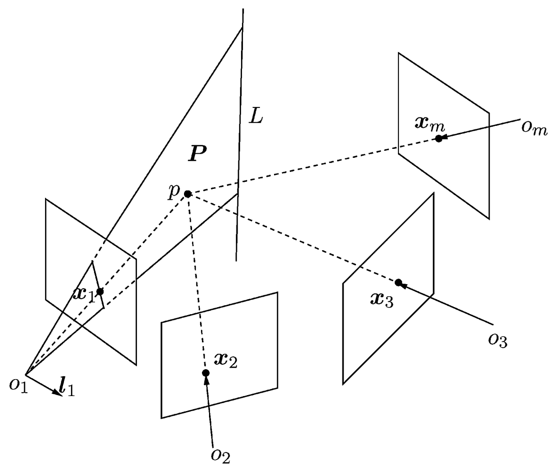

Figure 9. Generic configuration for the case $\text{rank}(M_{lp}) = 2$.

of these planes. Notice that, even in this case, adding more rows of line features to M will not be enough to uniquely determine L in 3-D. This is because the incidence condition for multiple line features requires that the rank of the associated matrix M_l be 1.[14] If we only require rank 2 for the overall matrix M, the line can be determined only up to a family of lines—intersections of the planes associated to all the line features—which all should intersect at the same point p.

Case (b), when the rank of M is 1, corresponds to a set of degenerate cases for which the incidence relationship between the point p and the line L will be violated. For example, it is direct to show that M_{lp} is of rank 1 if and only if all the vectors $R_i^{-1}\boldsymbol{x}_i, i = 2, \ldots, m$ are parallel to each other and they are all orthogonal to $\boldsymbol{l}_1$, and $R_i^{-1}T_i, i = 2, \ldots, m$ are also orthogonal to $\boldsymbol{l}_1$. That means all the camera centers lie on the same plane specified by o_1 and $\boldsymbol{l}_1$ and all the images $\boldsymbol{x}_2, \ldots, \boldsymbol{x}_m$ (transformed to the reference camera frame) lie on the same plane and are parallel to each other. For example, suppose that $m = 5$ and choose M to be:

$$M = \begin{bmatrix} \widehat{\boldsymbol{x}_2} R_2 \widehat{\boldsymbol{l}_1} & \widehat{\boldsymbol{x}_2} T_2 \\ \widehat{\boldsymbol{x}_3} R_3 \widehat{\boldsymbol{l}_1} & \widehat{\boldsymbol{x}_3} T_3 \\ \widehat{\boldsymbol{x}_4} R_4 \widehat{\boldsymbol{l}_1} & \widehat{\boldsymbol{x}_4} T_4 \\ \boldsymbol{l}_5^T R_5 \widehat{\boldsymbol{l}_1} & \boldsymbol{l}_5^T T_5 \end{bmatrix} \in \mathbb{R}^{10\times 4}. \quad (59)$$

The geometric configuration of the point and line features corresponding to the condition $\text{rank}(M) = 1$ is illustrated in Fig. 10.

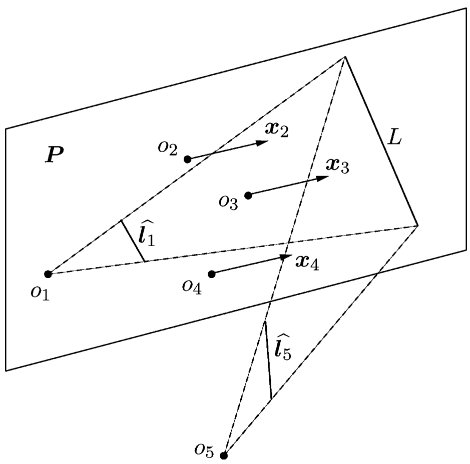

Figure 10. A degenerate configuration for the case $\text{rank}(M) = 1$: a line-point-point-point-line scenario.

Notice that in this case, we no longer have an incidence condition for the point features. However, one can view them as if they intersected at a point p at infinity. In general, we no longer have the incidence condition between the point p and the line L, unless both the point p and line L are in the plane at infinity in the first place. But since the rank condition is effective for line features, the incidence condition for all the line features still holds.

To summarize the above discussion, we see that the rank conditions indeed allow us to carry out meaningful global geometric analysis on the relationship among multiple point and line features for arbitrarily many views. There is no doubt that this extends existing methods based on multifocal tensors that can only be used for analyzing up to three views at a time. Since there is yet no systematic way to extend triple-wise analysis to multiple views, the multiple-view matrix seems to be a more natural tool for multiple-view analysis. Notice that the rank conditions imply all previously known multilinear constraints, but multilinear constraints do not necessarily imply the rank conditions. This is because the use of algebraic equations may introduce certain artificial degeneracies that make a global analysis much more complicated and sometimes even intractable. On the other hand, the rank conditions have no problem characterizing all the geometrically meaningful degeneracies in a multiple-view mixed-feature scenario. All the degenerate cases simply correspond to a further drop of rank for the multiple-view matrix.

6. Multiple-View Factorization

The unified formulation of the constraints among multiple images in terms of the rank conditions, allows us to tackle many problems in multiple-view geometry *globally*. More specifically, one can use these rank conditions to test whether or not a given set of features (points or lines) indeed satisfy certain incidence relations in $\mathbb{E}^3$. For example, whether a given set of point and line features indeed *correspond* one to another depends on whether all associated rank conditions are satisfied. The rank values can be used not only to detect outliers but also degenerate configurations. The rank conditions also allow us to *transfer* multiple features to a new view by using all features and incidence conditions among them simultaneously, without resorting to the 3-D structure of the scene. Hence the multiple-view matrix provides the geometric basis for any *view* *synthesis* or *image based rendering* techniques. We have already carried out some initial experiments exploring the utility of the rank constraints (Kosecka and Ma, 2002) and more thorough experiments and comparison with existing techniques is currently underway.

Due to the space limitations, here we only demonstrate the use of the rank condition on the structure and motion recovery problem from multiple views.[15] The proposed algorithm belongs to a class of global factorization based techniques for structure and motion recovery and enables simultaneous recovery of camera configurations (R_i, T_i)'s and structure of the scene. In case of perspective projection alternative projective factorization technique proposed by Triggs (1996). Since that algorithm is based on the less compact rank condition of matrix N_p (Eq. (11)), the problem is overparameterized and also requires nonunique initialization of projective scales. Due to the nature of the rank conditions presented here, we not only obtain minimal parameterization of the problem but can also simultaneously incorporate additional features. The incidence relations among given points and lines can now be explicitly taken into account when a global and consistent recovery of motion and structure takes place. In the following section, in addition to the presentation of the factorization based algorithm for the mixed feature case, we chose to examine the usefulness of the incidence relations in the multiple-view structure and motion recovery.

In order to outline the conceptual algorithm for mixed feature case, let us consider image of a cube as shown in Fig. 11. Let the jth corner p^j be the intersection of the three edges L^{1j}, L^{2j} and L^{3j}, $j = 1, 2, \ldots, 8$. Given m images of the cube, we have the multiple-view matrix M^j associated to p^j:

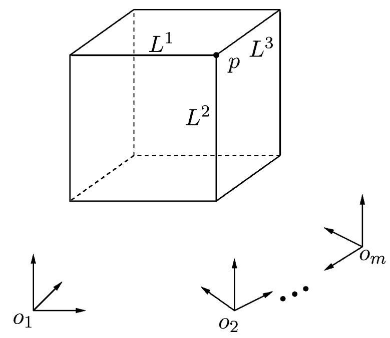

Figure 11. A standard cube. The three edges L^1, L^2, L^3 intersect at the corner p. The coordinate frames indicate that m images are taken at these vantage points.

$$M^j = \begin{bmatrix} \widehat{x_2^j} R_2 x_1^j & \widehat{x_2^j} T_2 \\ l_2^{1jT} R_2 x_1^j & l_2^{1jT} T_2 \\ l_2^{2jT} R_2 x_1^j & l_2^{2jT} T_2 \\ l_2^{3jT} R_2 x_1^j & l_2^{3jT} T_2 \\ \vdots & \vdots \\ \widehat{x_m^j} R_m x_1^j & \widehat{x_m^j} T_m \\ l_m^{1jT} R_m x_1^j & l_m^{1jT} T_m \\ l_m^{2jT} R_m x_1^j & l_m^{2jT} T_m \\ l_m^{3jT} R_m x_1^j & l_m^{3jT} T_m \end{bmatrix} \in \mathbb{R}^{[6(m-1)]\times 2}, \quad (60)$$

where $x_i^j \in \mathbb{R}^3$ means the image of the jth corner in the ith view and $l_i^{kj} \in \mathbb{R}^3$ means the image of the kth edge incident to the jth corner in the ith view. Lets see how can we use the rank condition $\text{rank}(M^j) = 1$ stated in Theorem 1 towards the structure and motion recovery. First note that $\alpha^j = [\lambda_1^j, 1]^T \in \mathbb{R}^2$ is in the kernel of M^j, where λ_1^j is the depth of p^j with respect to the reference frame. Hence in case the matrix M^j is known the depth of the point can be recovered. In addition to the multiple images x_1^j, x_2^j, x_3^j of the jth corner p^j itself, the extra rows associated to the line features $l_i^{kj}, k = 1, 2, 3, i = 1, 2, \ldots, m$ also contribute towards determination of the depth scale λ_1^j.

We can already see one advantage of the rank condition: It can simultaneously handle multiple incidence conditions associated to the same feature. In principle, by using (49) or Corollary 7, one can further take into account that the four vertices and edges on each face are coplanar. Since such incidence conditions and relations among points and lines occur frequently in practice, especially for man-made objects, such as buildings and houses, the use of multiple-view matrix for mixed features is going to improve the quality of the overall reconstruction by explicitly taking into account all incidence relations among features of various types.

In order to estimate α^j we need to know the matrix M^j, i.e., we need to know the motion (R_2, T_2), $\ldots, (R_m, T_m)$. From the geometric meaning of $\alpha^j = [\lambda_1^j, 1]^T$, α^j can be solved if we know only the motion (R_2, T_2) between the first two views, which can be initially estimated using the standard 8-point algorithm. Consider now that we have initial estimates of α^j's. Notice now that each row of M^j now becomes linear equation in (R_i, T_i) for $i = 2, \ldots, m$; e.g., first row will become $\lambda_1^j \widehat{x_2^j} R_2 x_1^j + \widehat{x_2^j} T_2 = 0$. Hence we can use the equations:

$$M^j \alpha^j = 0, \quad j = 1, 2, \ldots, 8 \tag{61}$$

to solve for the motions (again). Define the vectors:

$$\vec{R}_i = [r_{11}, r_{12}, r_{13}, r_{21}, r_{22}, r_{23}, r_{31}, r_{32}, r_{33}]^T \in \mathbb{R}^9$$

and $\vec{T}_i = T_i \in \mathbb{R}^3$, $i = 2, 3, \ldots, m$. Solving (61) for (R_i, T_i) is then equivalent to finding the solution to the following equations for $i = 2, 3, \ldots, m$:

$$P_i \begin{bmatrix} \vec{R}_i \\ \vec{T}_i \end{bmatrix} = \begin{bmatrix} \lambda_1^1 \widehat{x_i^1} * x_1^{1^T} & \widehat{x_i^1} \\ \lambda_1^1 l_i^{11T} * x_1^{1^T} & l_i^{11T} \\ \lambda_1^1 l_i^{21T} * x_1^{1^T} & l_i^{21T} \\ \lambda_1^1 l_i^{31T} * x_1^{1^T} & l_i^{31T} \\ \vdots & \vdots \\ \lambda_1^8 \widehat{x_i^8} * x_1^{8^T} & \widehat{x_i^8} \\ \lambda_1^8 l_i^{18T} * x_1^{8^T} & l_i^{18T} \\ \lambda_1^8 l_i^{28T} * x_1^{8^T} & l_i^{28T} \\ \lambda_1^8 l_i^{38T} * x_1^{8^T} & l_i^{38T} \end{bmatrix} \begin{bmatrix} \vec{R}_i \\ \vec{T}_i \end{bmatrix} = 0 \in \mathbb{R}^{48}, \tag{62}$$

where $A * B$ is the *Kronecker product* of A and B. In general, if we have more than 6 feature points (here we have 8) or equivalently 12 feature lines, the rank of the matrix P_i is 11 and there is a unique solution to $(\vec{R}_i, \vec{T}_i)$.

Let $\tilde{T}_i \in \mathbb{R}^3$ and $\tilde{R}_i \in \mathbb{R}^{3\times 3}$ be the (unique) solution of (62) in matrix form. Such a solution can be obtained numerically as the eigenvector of P_i associated to the smallest singular value. Let $\tilde{R}_i = U_i S_i V_i^T$ be the SVD of $\tilde{R}_i$. Then the solution of (62) in $\mathbb{R}^3 \times SO(3)$ is given by:

$$T_i = \frac{\operatorname{sign}\big(\det(U_i V_i^T)\big)}{\sqrt[3]{\det(S_i)}} \tilde{T}_i \in \mathbb{R}^3, \tag{63}$$

$$R_i = \operatorname{sign}\big(\det(U_i V_i^T)\big) U_i V_i^T \in SO(3). \tag{64}$$

In case the camera is not calibrated the scales with respect to the first frame λ_1^j can be initialized via projective reconstruction from two views and the rotation matrix constraints on $(\tilde{R}_i, \tilde{T}_i)$ will not be enforced. Additional normalization step of image coordinate is also recommended, so as to ensure better performance in the presence of noise.

We then have the following linear algorithm for motion and structure estimation from three views of a cube:

Algorithm 1 (*Motion and structure from mixed features*). *Given $m(=3)$ images $x_1^j, \ldots, x_m^j$ of $n(=8)$ points p^j, $j = 1, 2, \ldots, n$ (as the corners of a cube), and the images l_i^{kj}, $k = 1, 2, 3$ of the three edges intersecting at p^j, estimate the motions (R_i, T_i), $i = 2, 3, \ldots, m$ as follows*:

1. *Initialization*: $s = 0$
   (a) *Compute (R_2, T_2) using the 8-point algorithm for the first two views, using for example (Longuet-Higgins, 1981).*
   (b) *Compute $\alpha_s^j = [\lambda_1^j/\lambda_1^1, 1]^T$ where λ_1^j is the depth of the jth point relative to the first camera frame.*
2. *Compute $(\tilde{R}_i, \tilde{T}_i)$ as the eigenvector associated to the smallest singular value of P_i, $i = 2, 3, \ldots, m$.*
3. *Compute (R_i, T_i) from (63) and (64) for $i = 2, 3, \ldots, m$.*
4. *Compute new $\alpha_{s+1}^j = \alpha^j$ from (61). Normalize so that $\lambda_{1,s+1}^1 = 1$.*
5. *If $\|\alpha_s - \alpha_{s+1}\| > \epsilon$, for a pre-specified $\epsilon > 0$, then $s = s + 1$ and goto 2. Else stop.*

The camera motion is then the converged (R_i, T_i), $i = 2, 3, \ldots, m$ and the structure of the points (with respect to the first camera frame) is the converged depth scalar λ_1^j, $j = 1, 2, \ldots, n$.

We have a few comments on the proposed algorithm:

1. *Global scale.* The reason to set $\lambda_{1,s+1}^1 = 1$ is to fix the global scale. It is equivalent to putting the first point at a relative distance of 1 to the first camera center. The above algorithm is a straightforward modification of the algorithm proposed for the pure point case (Ma et al., 2001). All measurements of line features directly contribute to the estimation of the camera motion and the structure of the points. Throughout the algorithm, there is no need to initialize or estimate the 3-D parameters of lines.
2. *Line and planar cases.* Although the algorithm is based on the cube, considers only three views, and utilizes only one type of multiple-view matrix, it can be easily generalized to any other objects and arbitrarily many views whenever incidence conditions

among a set of point features and line features are present. One may also use it for rank conditions on different types of multiple-view matrices provided by Theorem 1. The reader may refer to (Ma et al., 2001) for the case when D_1 is chosen to be $\hat{\boldsymbol{l}}_1$, or the features are all coplanar. The same factorization works for these cases with only minor changes.

3. *Uncalibrated case.* Although the algorithm is proposed for the calibrated case, in the uncalibrated case, one may take any projective reconstruction to initialize α and factor. A detailed study on the uncalibrated case can be found in Ma et al. (2003).

The reader must be aware that the above algorithm is only *conceptual* (and naive in many ways). It by no means suggests that the resulting algorithm would work better in practice than some existing algorithms in every situation. The reason is, there are many possible ways to impose the rank conditions and each of them, although maybe algebraically equivalent, can have dramatically different numerical stability and sensitivity. To make the situation even worse, under different conditions (e.g., long baseline or short baseline), correctly imposing these rank conditions does require different numerical recipes.[16] A systematic characterization of numerical stability of the rank conditions remains largely open at this point. It is certainly the next logical step for future research.

7. Simulation Results

Given the above conceptual algorithm the following section presents simulation results in order to justify our intuition behind the suggested global approach to structure from motion recovery. While at this stage we do not make any optimality claims, due to the linear nature of the proposed algorithms, we will demonstrate the performance and dependency of the algorithm on types of features in the presence of noise.

7.1. Reconstruction With or Without Incidence Relations

The simulation parameters are as follows: the camera's field of view is 90°, image size is 500×500, everything is measured in units of the focal length of the camera, and features typically are suited with a depth variation is from 100 to 400 units of focal length away from the camera center, i.e., they are located in the truncated

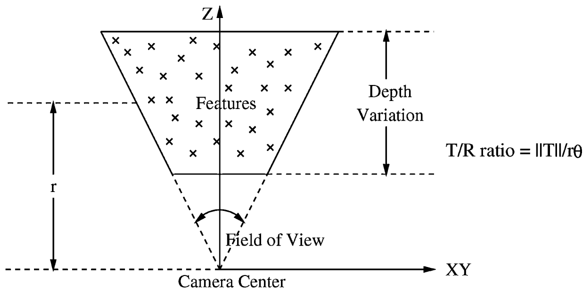

Figure 12. Simulation setup.

pyramid specified by the given field of view and depth variation (see Fig. 12). Camera motions are specified by their translation and rotation axes. For example, between a pair of frames, the symbol XY means that the translation is along the X-axis and rotation is along the Y-axis. If we have a sequence of such symbols connected by hyphens, it specifies a sequence of consecutive motions. We always choose the amount of total motion, so that all feature points will stay in the field of view for all frames. In all simulations, independent Gaussian noise with a standard deviation (std) given in pixels is added to each image point, and each image line is perturbed in a random direction of a random angle with a corresponding standard deviation given in degrees.[17] Error measure for rotation is $\arccos(\frac{tr(R\tilde{R}^T)-1}{2})$ in degrees where $\tilde{R}$ is an estimate of the true R. Error measure for translation is the angle between T and $\tilde{T}$ in degrees where $\tilde{T}$ is an estimate of the true T. Error measure for the scene structure is the percentage of $\|\alpha - \tilde{\alpha}\|/\|\alpha\|$ where $\tilde{\alpha}$ is an estimate of the true α.

We apply the algorithm to a scene which consists of (four) cubes. Cubes are good objects to test the algorithm since the relationships between their corners and edges are easily defined and they represent a fundamental structure of many objects in real-life. It is certainly a first step to see how the multiple-view matrix based approach is able to take into account point and line features as well as their inter-relationships to facilitate the overall recovery. The length of the four cube edges are 30, 40, 60 and 80 units of focal length, respectively. The cubes are arranged so that the depth of their corners ranges from 75 to 350 units of focal length. The three motions (relative to the first view) are an XX-motion with -10 degrees rotation and 20 units translation, a YY-motion with 10 degrees rotation and 20 units translation and another YY-motion with -10 degrees rotation and 20 units translation, as shown in Fig. 13.

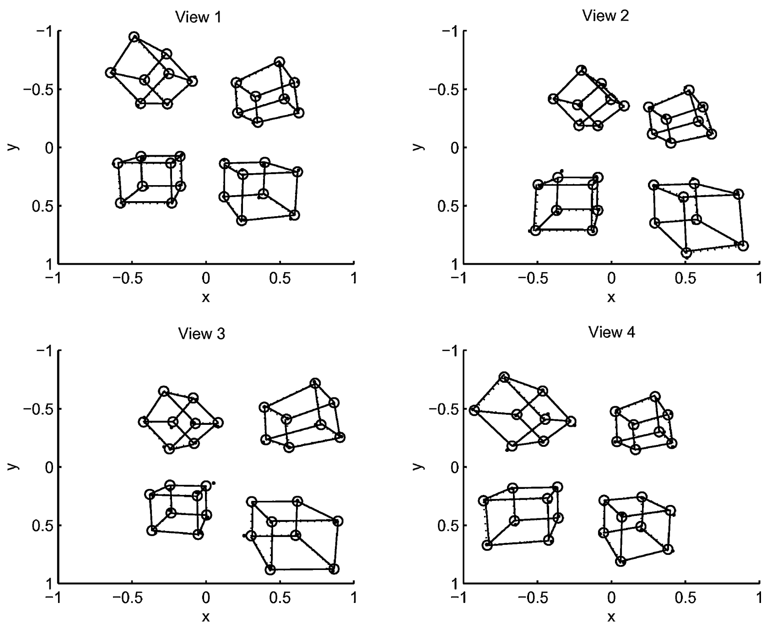

Figure 13. Four views of four 3-D cubes in (normalized) image coordinates. The circle and the dotted lines are the original images, the dots and the solid lines are the noisy observations under 5 pixels noise on point features and 0.5 degrees noise on line features.

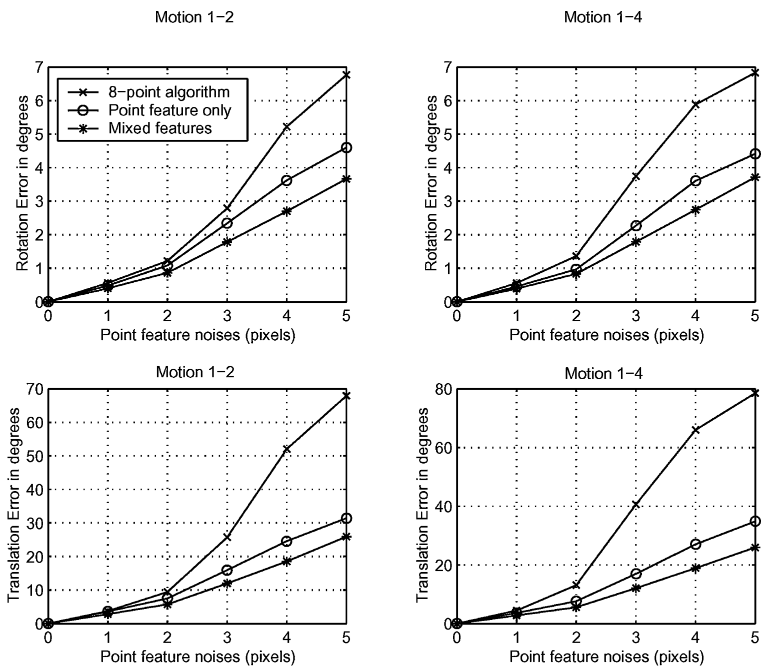

Figure 14. Motion estimates error versus level of noises. "Motion x-y" means the estimate for the motion between image frames x and y. Since the results are very much similar, we only plotted "Motion 1–2" and "Motion 1–4."

We run the algorithm for 1000 trials with the noise level on the point features from 0 pixel to 5 pixels and a corresponding noise level on the line features from 0 degree to 1 degrees. Relative to the given amount of translation, 5 pixels noise is rather high because we do want to compare how all the algorithms perform over a large range of noise levels. The results of the motion estimate error and structure estimate error are

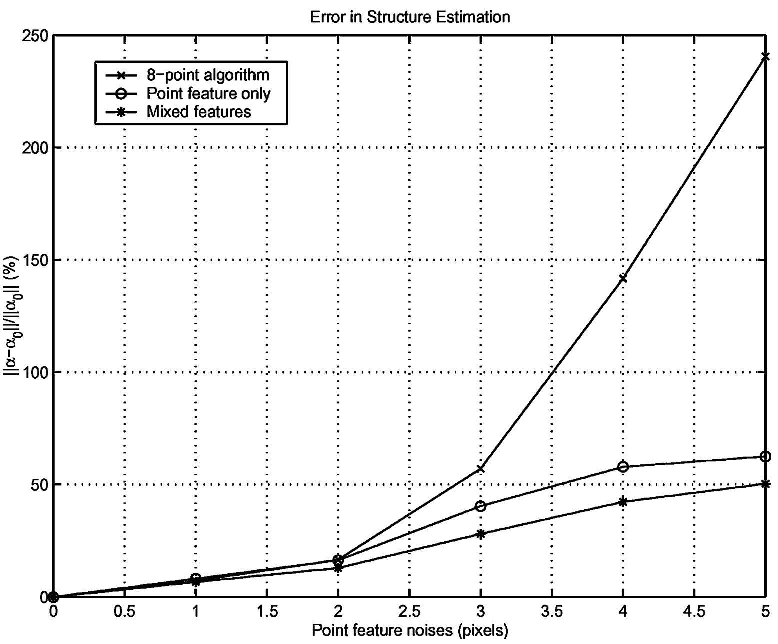

Figure 15. Structure estimates error versus level of noises. Here α_0 represents the true structure and α the estimated.

given in Figs. 14 and 15 respectively. The "Point feature only" algorithm essentially uses the multiple-view matrix M in (61) without all the rows associated to the line features; and the "Mixed features" algorithm uses essentially the same M as in (61). Both algorithms are initialized by the standard 8-point algorithm. The "Mixed features" algorithm gives a significant improvement in all the estimates as a result of the use of both point and line features in the recovery. Also notice that, at a high noise levels, even though the 8-point algorithm gives rather off initialization values, the two iterative algorithms manage to converge back to reasonable estimates.

8. Conclusions and Future Work

This paper has proposed a unified paradigm which synthesizes results and experiences in the study of multiple-view geometry of mixed features. It is shown that all incidence relations among multiple images of a point or a line are captured through certain rank conditions on a so-called multiple-view matrix. It is proven that the rank conditions are a super set of all previously known multiple-view constraints. Relationships among these constraints become rather simple consequences from the rank conditions. To a large extent, the theory developed in this paper simplifies and unifies the study of multiple-view geometry and enables us to carry out meaningful geometric analysis for arbitrarily many images, without going through a pairwise, triple-wise or quadruple-wise analysis. Compared to conventional multiple-view analysis based on trifocal tensors, the multiple-view matrix based approach clearly separates meaningful geometric degeneracies from algebraic degeneracies which may be artificially introduced by the use of algebraic equations or tensors. In particular, as shown in this paper, any configuration which causes a further drop of rank in the multiple-view matrix *exactly* corresponds to certain global geometric degeneracy. Combined with generalized rank conditions for a space of arbitrary dimension, i.e. rank conditions for projections from $\mathbb{R}^n$ to $\mathbb{R}^k$ with $k < n$ (see Fossum et al., 2001), results in this paper give rise to a coherent but simple geometric theory that is genuine for multiple images, for both static and dynamical scenes.

The approach proposed in this paper aims to study multiple-view geometry from a new perspective. It intends to provide a clear theoretical basis for any further development of algorithms, so that a systematic evaluation of algorithms will be possible. Most importantly, we hope to identify difficulties in multiple-view geometry that are caused by geometric configurations of the given data or by numerical methods adopted in a particular algorithm. The linear algorithms given in this paper and others (Ma et al., 2001) only show a straight-forward (hence naive) way of using these rank conditions. There are many possible ways to improve them: 1. One can use better error measures in the 2-D image to recover the motion and structure optimally subject to the rank condition; 2. Slight change of the algorithm may handle occlusions; 3. Better numerical methods should be investigated on how to impose the rank condition; and so on. Currently, we are developing practical algorithms based on the multiple-view matrix for various purposes: feature matching, feature transfer (or view synthesis), and image-based modeling and rendering. Readers who are interested in more recent development along this line of work may refer to the recent book (Ma et al., 2003). While we are still in the process of investigating the full potential of this new approach and developing practical algorithms for real applications, there are plenty of reasons for us to believe that we are still at a *very early* stage of understanding the full extent of multiple-view geometry: either its theory or its practice.

Notes

1. $SL(3)$ is the *special linear group*, i.e., the group of all 3×3 matrices with determinant 1. In computer vision literature, $K(t)$ is typically chosen to be an upper triangular matrix. That assumption does not make the problem at hand any easier.
2. $SE(3)$ is the group of all *special Euclidean* transformations, which preserve the Euclidean metric of $\mathbb{E}^3$.
3. $SO(3)$ is *special orthogonal group*, i.e., the group of all 3×3 rotation matrices.
4. In fact, there is some redundancy using $\hat{l}$ to describe the plane: the three column (or row) vectors in $\hat{l}$ are not linearly independent. They only span a two-dimensional space. However, we here use it anyway to simplify the notation.
5. Depending on the context, the reference frame could be either an Euclidean, Affine or Projective reference frame. In all cases the projection matrix for the first image becomes $[I,\ 0] \in \mathbb{R}^{3\times4}$.
6. According to Eq. (3), they are in fact (KR, KT). We here use (R, T) to simplify the notation.
7. That is any other polynomials that the images satisfy must contain these polynomials as factors.
8. However, we will show that quadrilinear constraints are actually useful (only) in the case in which a family of lines intersects at a point.
9. Again, here this statement is regarding multiple *corresponding* images of a line. As we will soon see, however, new constraints may arise when the images do not have to correspond to the same line in 3-D space.
10. This is due to the redundancy of using $\hat{u}$ for the orthogonal supplement of $u \in \mathbb{R}^3$.
11. Hence it is a constraint that is *not* in any of the multilinear (or multifocal) constraint lists previously studied in the computer vision literature. For a list of these multilinear constraints (see Hartley and Zisserman, 2000).
12. In fact, there are many equivalent matrix representations for D_i and $D_i^{\perp}$. We choose $\widehat{\boldsymbol{x}_i}$ and $\widehat{\boldsymbol{l}_i}$ here because they are the simplest forms representing the orthogonal subspaces of $\boldsymbol{x}_i$ and $\boldsymbol{l}_i$ and also linear in $\boldsymbol{x}_i$ and $\boldsymbol{l}_i$ respectively.
13. In fact, this is quite expected: While the rank condition geometrically corresponds to the incidence condition that lines intersect at a point and that planes intersect at a line, incidence condition that three-dimensional subspaces intersect at a plane is a void condition in $\mathbb{E}^3$.
14. Matrix M_l is obtained from M by extracting the line measurements only.
15. The reader is referred to Ma et al. (2001) for details about applications to other problems in multiple-view geometry.
16. This is true even for the standard 8-point algorithm in the two view case.
17. Since line features can be measured more reliably than point features, lower noise level is added to them in simulations.

References

Avidan, S. and Shashua, A. 1998. Novel view synthesis by cascading trilinear tensors. *IEEE Transactions on Visualization and Computer Graphics (TVCG)*, 4(4).

Faugeras, O., Luong, Q.-T., and Papadopoulo, T. 2001. *Geometry of Multiple Images*. The MIT Press.

Faugeras, O. and Mourrain, B. 1995. On the geometry and algebra of the point and line correspondences between *N* images. In *Proceedings of Fifth International Conference on Computer Vision*, IEEE Comput. Soc. Press: Cambridge, MA, USA, pp. 951–956.

Hartley, R. 1994. Lines and points in three views—a unified approach. In *Proceedings of 1994 Image Understanding Workshop*, OMNIPRESS: Monterey, CA, USA, pp. 1006–1016.

Hartley, R. and Zisserman, A. 2000. *Multiple View Geometry in Computer Vision*. Cambridge.

Heyden, A. and Åström, K. 1997. Algebraic properties of multilinear constraints. *Mathematical Methods in Applied Sciences*, 20(13):1135–1162.

Kahl, F. and Heyden, A. 1999. Affine structure and motion from points, lines and conics. *International Journal of Computer Vision*, 33(3):163–180.

Kosecka, J. and Ma, Y. 2002. Introduction to multiview rank conditions and their applications: A review. *Tyrrhenian International Workshop on Digital Communications (IWDC 2002)*.

Kruppa, E. 1913. Zur ermittlung eines objecktes aus zwei perspektiven mit innerer orientierung. *Sitz.-Ber. Akad. Wiss., Math. Naturw., Kl. Abt. IIa*, 122:1939–1948.

Liu, Y., Huang, T., and Faugeras, O. 1990. Determination of camera location from 2-D to 3-D line and point correspondences. *IEEE Transactions on PAMI*, 28–37.

Longuet-Higgins, H.C. 1981. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135.

Ma, Y., Huang, K., Vidal, R., Košecká, J., and Sastry, S. 2001. Rank conditions of multiple-view matrix in multiple view geometry.

UIUC, CSL Technical Report, UILU-ENG 01-2214 (DC-220), June 18.

Ma, Y., Soatto, S., Košecká, J., and Sastry, S. 2003. *An Invitation to 3-D Vision: From Images to Geometric Models.* Interdisciplinary Applied Mathematics. Springer-Verlag.

Fossum, R., Huang, K., Ma, Y. 2001. General rank conditions in multiple view geometry. UIUC, CSL Technical Report, UILU-ENG 01-2222 (DC-203).

Quan, L. and Kanade, T. 1996. A factorization method for affine structure from line correspondences. In *Proceedings of the CVPR*, pp. 803–808.

Shashua, A. 1994. Trilinearity in visual recognition by alignment. In *Proceedings of ECCV*, Vol. I, Springer-Verlag, pp. 479–484.

Shashua, A. and Wolf, L. 2000. On the structure and properties of the quadrifocal tensor. In *Proceedings of ECCV*, Vol. I, Springer-Verlag, pp. 711–724.

Spetsakis, M. and Aloimonos, Y. 1990. Structure from motion using line correspondences. *International Journal of Computer Vision*, 4(3):171–184.

Taylor, C.J. and Kriegman, D. 1995. Structure and motion from line segments in multiple images. *IEEE Transactions of Pattern Analysis and Machine Intelligence*, 17(11).

Tomasi, C. and Kanade, T. 1992. Shape and motion from image streams under orthography. *Intl. Journal of Computer Vision*, 9(2):137–154.

Triggs, B. 1995. Matching constraints and the joint image. In *Proceedings of Fifth International Conference on Computer Vision*, IEEE Comput. Soc. Press: Cambridge, MA, USA, pp. 338–343,

Triggs, B. 1996. Factorization methods for projective structure and motion. In *Proceedings of 1996 Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE Comput. Soc. Press: San Francisco, CA, USA, pp. 845–851.