

# Diagram Techniques for Multiple View Geometry

Alberto Ruiz

DIS, Universidad de Murcia, Spain

<http://dis.um.es/~alberto>

Pedro E. Lopez-de-Teruel

DITEC, Universidad de Murcia, Spain

<http://ditec.um.es/~pedroe>

## Abstract

*Multilinear algebra is a powerful theoretical tool for visual geometry, but widespread usage of traditional typographical notation often hides its conceptual elegance and simplicity. As demonstrated in other scientific fields, we can take full advantage of multilinear methods using graphical notation. In this paper we adapt standard tensor diagrammatic techniques to the specific requirements of visual geometry, so that geometric relations are represented by circuits which can be manipulated using simple rules.*

*The advantages of this approach are illustrated in several constructions, including straightforward derivations of the standard multiview relations (Fundamental Matrix, Trifocal and Quadrifocal Tensors), and nearly mechanical procedures for camera extraction.*

## 1. Introduction

The geometry of multiple images can be naturally described in terms of multilinear relations [1, 2]. Tensors and Exterior Algebra [3, 4, 5] are appropriate tools for the study of projective entities represented as subspaces [6, 7, 8]. Unfortunately, it is often difficult to take full advantage of the power and elegance of multilinear methods using standard typographical notation.

In contrast, graphical representations are remarkably useful for visualization and manipulation of complex mathematical concepts, and have widespread usage in several scientific fields, specially in Physics [9, 10, 11, 12, 13, 14, 15]. Feynman Diagrams, Spin Networks, Trace Diagrams, and similar methods provide deep insight into mathematical structure, exposing interesting symmetries and manipulation possibilities.

In this work we develop a diagrammatic approach suitable to analyze and solve a wide range of visual geometry problems. This technique has many advantages: for instance, the multiview tensors can be directly derived from the geometrical setting using meaningful building blocks. Furthermore, visualization of the internal structure of the tensors suggests effective procedures for extraction of com-

patible cameras. Diagrams can be simplified or rearranged using straightforward transformation rules, much like working with electronic circuits or logic gates.

The paper is organized as follows. Sections 2 and 3 introduce the diagrammatic notation and appropriate manipulation rules for visual geometry. Then we apply the proposed technique to obtain the multiview tensors (Section 4), and to extract compatible cameras from their internal structure (Section 5). The computational advantages of the approach are discussed in Section 6. The last section contains some concluding remarks.

## 2. Diagrammatic Notation

We adopt standard diagrammatic conventions [11, 10], with minor modifications to easily keep track of the geometric role of the different objects.

### 2.1. Tensors

For our purposes, a tensor of rank  $r$  is a multilinear function of  $r \geq 0$  arguments. The arguments can be either vectors or covectors (scalar-valued linear functions), from possibly different vector spaces. Because of linearity, we can partially apply  $0 \leq m \leq r$  freely chosen inputs to obtain a  $(r - m)$ -rank tensor. The role of the arguments is not fixed: depending on how we use it, a vector input can play the role of a covector output and vice versa. Rank  $r$  tensors are represented by an  $r$ -dimensional array of coordinates. Coordinates that transform as vectors in a change of basis are called *contravariant*, while *covariant* coordinates transform as linear functions. Application of arguments and composition of functions are equivalent concepts, carried out by *contraction* of coordinates.

In diagrammatic notation tensor expressions are represented by graphs whose nodes are tensors and the edges are contractions. The degree of a node is the rank of the corresponding tensor. The number of open ‘legs’ in a diagram is the rank of the resulting tensor. The standard notation typically uses a common node shape for all kind of tensors, and the type of coordinate (covariant or contravariant) is labeled by an arrow (Fig. 1).

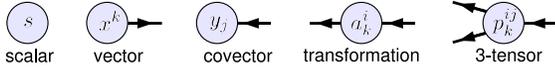


Figure 1. Standard tensor notation.

For clarity we use different shapes for the ‘slots’ of a tensor: contravariant coordinates have a convex angle and covariant coordinates have a concave one. Invariants have ‘straight’ sides. Fig. 2 shows node shapes of typical objects.

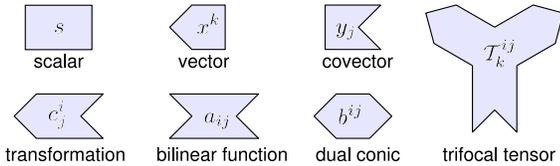


Figure 2. Tensor shape convention.

Using this notation open edges and arrows are usually not necessary to identify the type of a tensor. Contractions are represented by joining complementary ‘slots’ (Fig. 3).

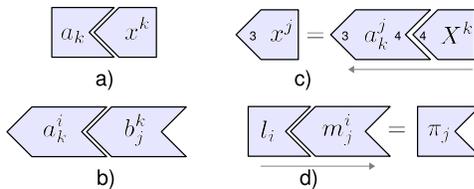


Figure 3. Contraction examples: a) Application of a linear function on a vector. b) Composition of transformations. c) Transformation of vectors (e.g. forward camera projection of points). d) Transformation of covectors (e.g., camera line resection).

The direct tensor product (without contraction) is represented by simple juxtaposition of nodes (Fig. 4).

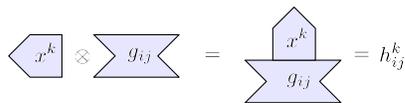


Figure 4. The tensor product.

If desired, space dimensions can be made explicit as small numbers, as in Fig. 3.c.

## 2.2. Exterior Algebra

Projective objects such as points, lines or planes in  $\mathbb{P}^n$  are subspaces of  $\mathbb{R}^{n+1}$ . Subspaces of dimension  $r$  are conveniently represented by  $r$ -vectors, a special type of tensor constructed using the exterior product, which is just the antisymmetrization of the direct tensor product. Antisymmetrization is graphically represented using Penrose’s crossing line convention [10] (Fig. 5). The full antisymmetric contravariant tensor  $\epsilon$  (also known as Levi-Civita sym-

bol) is the  $n$ -vector encoding the whole space. It is usually represented in the diagrams as a small circle.

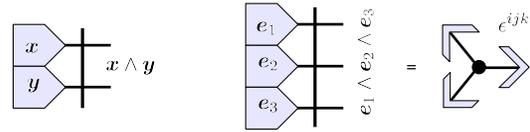


Figure 5. The exterior product (left) and the full antisymmetric contravariant tensor in  $\mathbb{R}^3$  (right).

A linear transformation  $A$  on vectors induces a transformation  $A^{(p)}$  on subspaces. It is achieved by applying the transformation to all the slots of the  $p$ -vector, as shown in Fig. 6. For instance, if  $A$  is a camera matrix,  $A^{(2)}$  is the corresponding forward projection for lines.

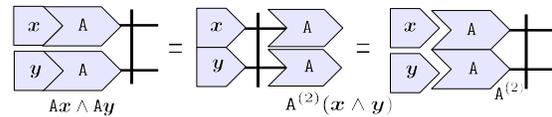


Figure 6. Transformation of 2-vectors.

Similarly, an inner product  $g_{ij}$  in a vector space (shaded in the diagrams) induces an inner product on multivectors (subspaces):

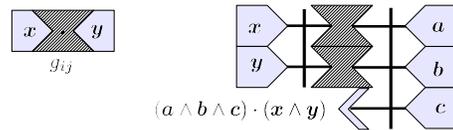


Figure 7. Inner product of multivectors.

This operation has an interesting geometric interpretation. If  $U$  is a  $p$ -vector and  $V$  is a  $q$ -vector,  $q \leq p$ , then the resulting  $(p - q)$ -multivector  $U \cdot V$  is the orthogonal complement of the projection of  $V$  onto  $U$ . This property will be frequently used later. (It is also known as the *contractive* inner product in geometric algebra [16].)

## 2.3. Dual

The dual  $*x$  of a  $p$ -vector  $x$  is the inner product of  $x$  with the  $n$ -vector representing the whole space  $\mathbb{R}^n$  (Fig. 8). Its rank is  $n - p$ . Some objects (e.g. lines in the plane or planes in space) are represented in a more economical way in dual form.

Computing the dual requires an inner product to choose an orthogonal direction in the complementary subspace. In projective geometry this particular direction is immaterial, but the inner product cannot be neglected because we must keep track of the covariant/contravariant nature of all tensor slots. To simplify notation the inner product required by dualization will be embedded in the full covariant antisymmetric tensor (Fig. 9).

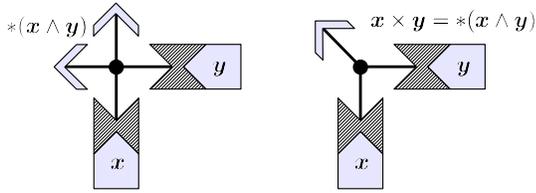


Figure 8. Dual of a 2-vector in  $\mathbb{R}^4$  (left), and  $\mathbb{R}^3$  (right), which is the *cross* product. Note that the exterior product is automatically performed by the dualization operation.

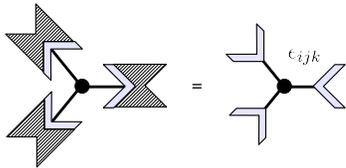


Figure 9. Dualization operator in  $\mathbb{R}^3 \simeq \mathbb{P}^2$ .

### 3. Graph reduction rules

#### 3.1. Basic rules

Since dualization is an involution, the composition of two full antisymmetric tensors is the identity for multivectors (Fig. 10).

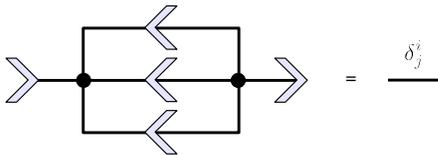


Figure 10. First graph reduction rule, for 1-vectors in  $\mathbb{P}^3$ .

Furthermore, the transformation of the whole space with any non degenerate transformation has no effect (modulo a projectively irrelevant scale factor equal to the determinant of the transformation). This is the basis of a second reduction rule (Fig. 11).

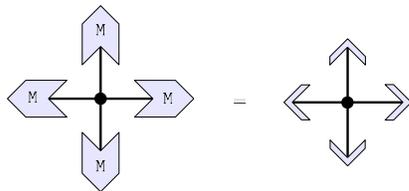


Figure 11. Second graph reduction rule, for  $\mathbb{P}^3$ .

The above rules can be used to obtain useful results in elementary linear algebra. For instance, Fig. 12 shows a diagram version of Cramer's rule for the inverse of a (homogeneous) transformation (Fig. 12).

#### 3.2. Preimage of a transformation

More interestingly, the inversion scheme in Fig. 12 also makes sense when the transformation  $M$  maps vectors to

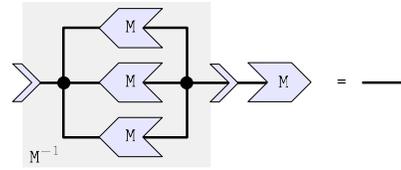


Figure 12. Inversor circuit.

spaces of lower dimension (e.g., a camera). In this case the additional unconnected slots in the final dualization step give rise to  $p$ -vectors of higher dimension, effectively obtaining the *preimage* subspace.

For example, Fig. 13 shows a circuit for reprojection of image points. The preimage transformation, denoted by  $M^{\leftarrow}$ , elegantly obtains the 3D line as a 2-vector.

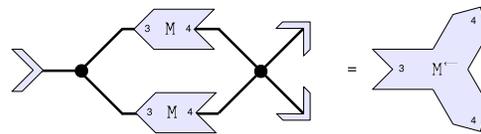


Figure 13. Preimage transformation of camera  $M$ : Point-to-ray reprojection.

Similarly, line reprojection can be expressed as the preimage transformation for 2-vectors (Fig. 14):

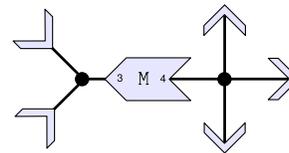


Figure 14. Line-to-plane camera reprojection.

Since 2D lines and 3D planes are more economically represented in dual form, line reprojection typically uses just  $M^T$ . In general, linear transformations work in the 'opposite' direction on dual objects.

The null-space of a transformation (e.g., the camera center) is the preimage of the trivial (zero) subspace (Fig. 15).

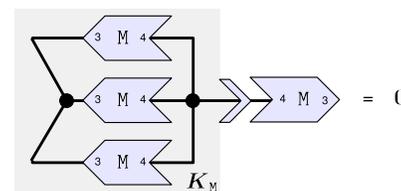


Figure 15. Null-space of a transformation.

#### 3.3. Rank-deficient transformations

If the transformation is not of full rank, the second graph reduction rule (Fig. 11) cannot be applied. The whole space will be transformed into a null multivector (there are not

enough linear independent components in the result). In this case we can apply a more general reduction rule shown in Fig. 16. It is based on two alternative expressions for the dual of the null-space.

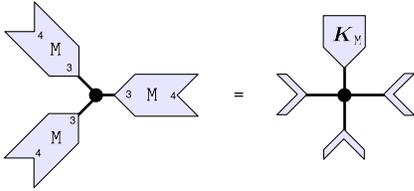


Figure 16. The general elimination rule, applied to  $M : \mathbb{R}^4 \rightarrow \mathbb{R}^3$ , with  $MK_M = 0$ .

Finally, if we have a rank deficient transformation  $M : \mathbb{R}^n \rightarrow \mathbb{R}^n$  the diagram in Fig. 15 cannot be used. In this case the null-space can be computed as in Fig. 17. This construction is again based on the fact that the whole space is transformed into a null multivector.

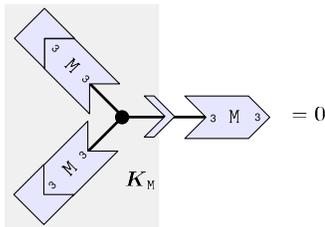


Figure 17. Null-space of a rank-deficient transformation. We may connect any covector (not through the null-space) to the left M's.

### 3.4. Geometric constructions

Exterior Algebra's uniform treatment of points, lines, planes, etc., in every dimension, is extremely convenient. Many useful geometric constructions can be ultimately described in terms of contractions with the full antisymmetric tensor. The dual of the union of subspaces is obtained by the covariant  $\epsilon$  (similar to the NOR logical gate), and the intersection of duals is obtained by the contravariant one (there is an analogy with De Morgan's laws). This approach is found in the literature under different terms: Double (Grassmann-Cayley) Algebra, *join* and *meet* operators, etc. [7, 2, 16]. Fig. 18 shows diagrams for the plane defined by a line and a point, and for the point of intersection of a plane and a line.

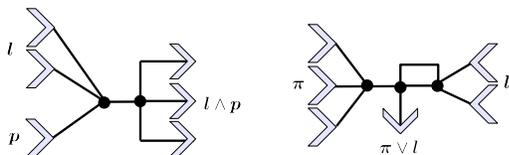


Figure 18. The *join* and *meet* operations in  $\mathbb{P}^3$ .

Complex geometric constructions can be described in terms of meaningful building blocks and then simplified us-

ing the above graph reduction rules. For instance, Fig. 19 shows a possible circuit for triangulation of points  $p$  and  $q$  imaged respectively by cameras  $M$  and  $N$ .

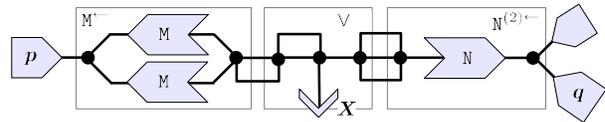


Figure 19. Triangulation circuit based on direct composition of geometrically meaningful building blocks.

This diagram is based on naive combination of the preimage of  $p$  (as in Fig. 13), the preimage of any line passing through  $q$  (as in Fig. 14), and the intersection (*meet*) circuit in Fig. 18. We observe that the first graph reduction rule (Fig. 10) can be applied twice to cancel out four redundant antisymmetric nodes. Fig. 20 shows the simplified circuit and a geometric interpretation.

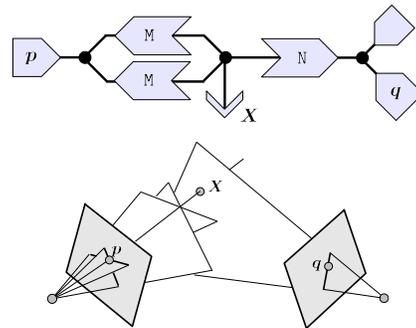


Figure 20. Simplified triangulation circuit. It can be interpreted as the intersection of three planes.

Interestingly, by reversing the input/output role of the  $X$  slot we get the homography induced by a plane (Fig. 21).

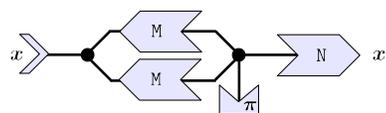


Figure 21. Homography between two views induced by a plane.

## 4. The Multiview Tensors

In this section we will apply the diagrammatic approach to study the multilinear relations among multiple images. The key fact is that the 3 degrees of freedom of a point in space, which can be obtained from 3 'half-points' distributed among two or three images, can be combined to predict the image of the point in any other view without explicit 3D reconstruction.

### 4.1. Two views

Given two views obtained by cameras  $M$  and  $N$ , the Fundamental Matrix gives the image  $l$  (in dual form) in the sec-

ond view of the reprojected ray (preimage) of a point  $x$  in the first view (Fig. 22).

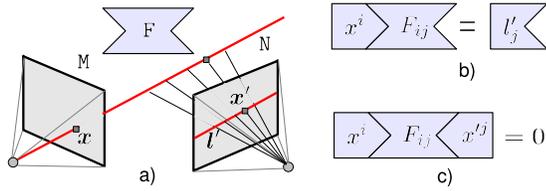


Figure 22. a) Stereo geometry. b) Epipolar line computation. c) Epipolar constraint.

This operation can be implemented as the composition of the preimage of M, the  $N^{(2)}$  transformation of 2-vectors, and a final dualization step. The circuit for  $F = *(N^{(2)}M^-)$  is shown in Fig. 23.

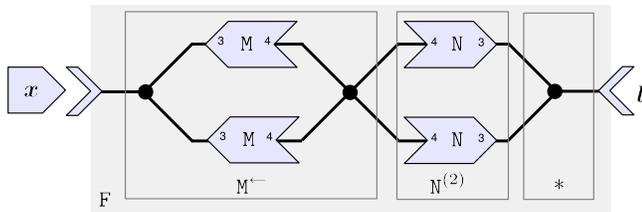


Figure 23. The internal structure of the Fundamental Matrix.

The symmetry of this construction shows that the Fundamental Matrix works automatically in both directions.

### 4.2. Three views

The image  $l$  of a 3D line in a view is determined by the images  $l'$  and  $l''$  of that line in two other views (Fig. 24). This relation is captured by the Trifocal Tensor  $\mathcal{T}$ .

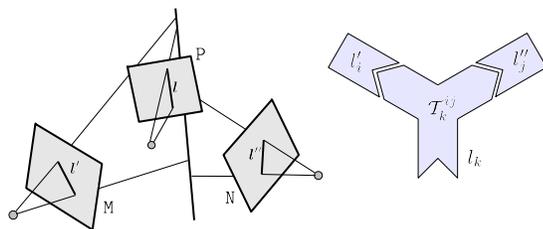


Figure 24. Trifocal geometry.

A diagram for  $\mathcal{T}$  is shown in Fig. 25. The circuit computes the 3D line as the intersection of two reprojected planes from cameras M and N, and obtains its image (in dual form) in the third view using  $P^{(2)}$ . The structure of  $\mathcal{T}$  shows the different roles of the views associated to  $l'$  and  $l''$  and the distinguished view associated to  $l$ .

Partial application of one argument induces two kind of transfer homographies, as illustrated in Fig. 26. The digrammatic convention for the slots immediately suggests consistent usages of the tensor.

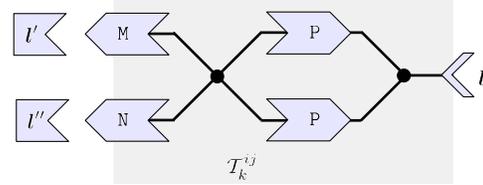


Figure 25. The internal structure of the Trifocal Tensor.

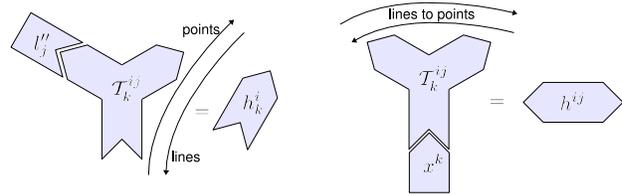


Figure 26. Transfer homographies arising from the Trifocal Tensor

Incidence conditions (e.g. required for the estimation of  $\mathcal{T}$  from point or line correspondences) can also be easily constructed (Fig. 27). The full antisymmetric tensor is used here as an homogeneous equality detector.

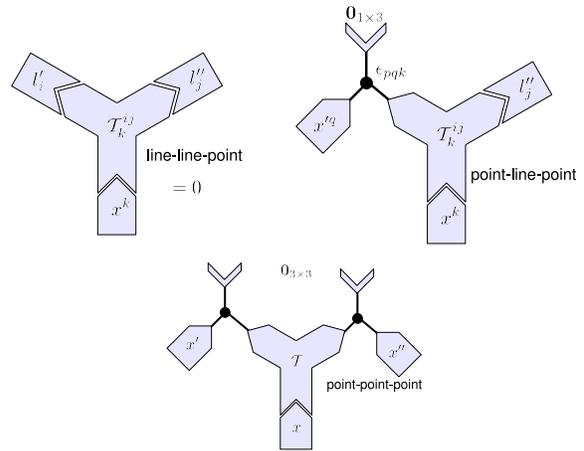


Figure 27. Some trifocal incidence relations.

### 4.3. Four views

The image of a point in a fourth view is determined by three ‘half images’ of that point in three different views. This relation is captured by the Quadrifocal Tensor  $\mathcal{Q}$ . A diagram for  $\mathcal{Q}$  can be constructed by taking the intersection of the planes reprojected from lines going through the point in three views, and projecting the reconstructed 3D point in the fourth view (Fig. 28).

The internal structure of  $\mathcal{Q}$  shows that the Fundamental Matrix and the Trifocal Tensor are just particular cases, in which some of the cameras appear more than once. For two views, two ‘half-points’ are taken from the same camera ( $M = A = B$ ), and the third one is taken from the second camera, which is also the view in which we project the 3D

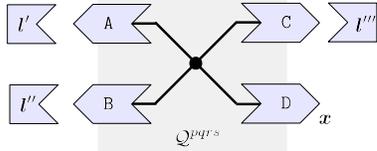


Figure 28. The internal structure of the Quadrifocal Tensor.

point ( $N = C = D$ ). In the three-view case the image of the 3D point is obtained on one of the views which provided one half point.

## 5. Camera Extraction

The internal structure of the multiview tensors can be easily manipulated for extraction of compatible cameras using graph reduction rules.

### 5.1. Two views

Consider the internal structure of the Fundamental Matrix (Fig. 23). Any full-rank transformation of the scene will cancel out in the central  $\epsilon$ , in accordance with the projective ambiguity of 3D reconstruction. Therefore, we are free to arbitrarily choose the first camera (typically  $M = [I|0]$ ). In order to get a cancellation configuration we connect  $M$  on the left slot and revert the dualization on the  $N$  side (Fig. 29).

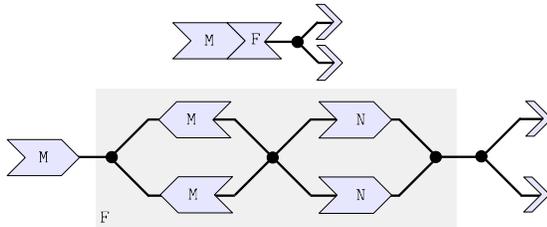


Figure 29. The structure of  $\epsilon$ FM.

If  $M$  were invertible we could apply the first graph reduction rule (Fig. 10), leaving just the second camera (in  $N^{(2)}$  form). Since cameras map to a lower dimension space we must apply instead the general elimination rule (Fig. 16), obtaining the result shown in Fig. 30.

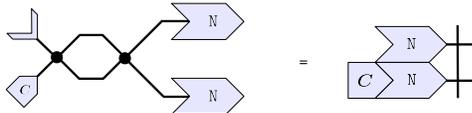


Figure 30. Simplification of Fig. 29.  $C$  is the center of  $M$ .

This can be interpreted as a ‘semicamera’, which produces epipolar lines instead of image points (Fig. 31).

At first sight it seems difficult to extract a compatible camera from the above antisymmetrized mixture. However, it can be easily done by applying the restricted dualization operation described in Fig. 7. We connect one of the outputs

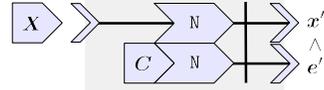


Figure 31.  $\epsilon$ FM obtains epipolar lines.

to an arbitrary covector  $v$  (Fig. 32), obtaining a point  $\tilde{x}'$  in the epipolar line which is different from  $v$ .

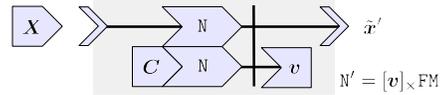


Figure 32. The ‘covering’ step.

The epipole  $e' = v^T$  is typically used to guarantee that the obtained point is never  $e'$  itself. This is the only condition for a compatible camera  $N'$ .

(The epipole is just the right null-space of  $F$ . It can be obtained using the method in Fig. 17, which can be also interpreted as the intersection of two epipolar lines (Fig. 33).)

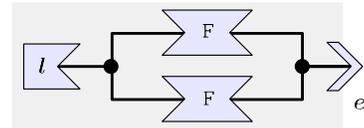


Figure 33. A circuit for the epipole.

In a sense,  $N'$  tries to imitate the true  $N$  by mapping the 3D space into the second view indirectly through the first one. Unfortunately, in this route  $N'$  suffers an additional rank loss: the null-space of  $N'$  contains the centers of both cameras. The baseline cannot be projected (Fig. 34).

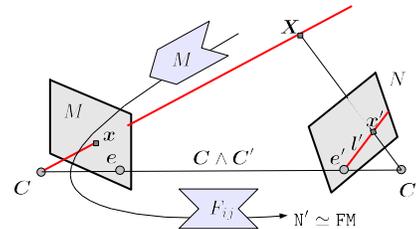


Figure 34. Double rank loss in camera extraction.

To solve this problem we additively combine  $N'$  with a rank-1 auxiliary camera which maps  $C$  to its image in the second view (the epipole  $e'$ ). A complete diagram for camera extraction from  $F$  is shown in Fig. 35. This is actually a diagrammatic version of the standard expression for the canonical cameras  $M = [I|0]$  and  $N = [[e']_x F | e']$ .

### 5.2. Three views

Compatible cameras can also be extracted from the Trifocal Tensor  $\mathcal{T}$  by a sequence of graph manipulation steps based on intuitive geometric reasoning. The structure of this

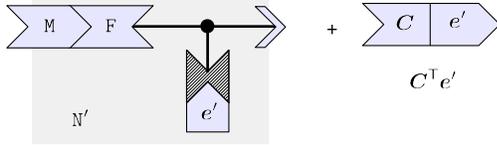


Figure 35. Complete diagram for a second camera  $N$  compatible with the Fundamental Matrix.

tensor is particularly adequate for camera extraction due to the direct availability of transfer homographies.

First note that  $\mathcal{T}$  can be used to simulate the behavior of the Fundamental Matrix of any image pair. For example, we can compute epipolar lines by joining the images in the second view of a point  $x_1$  in the first view transferred through two different lines  $a$  and  $b$  in the third view (Fig. 36).

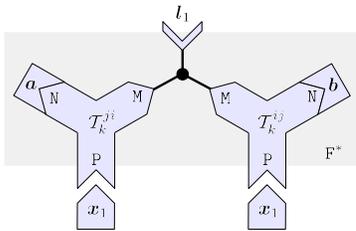


Figure 36. Simulation of a Fundamental Matrix from  $\mathcal{T}$ .

This operation in itself does not generate  $F_{12}$ , since it is quadratic in  $x$ , but it can be used to compute the epipole  $e'$  by intersection of the epipolar lines induced by two points. Then  $F_{12}$  can be obtained as an arbitrary transfer joined to the epipole.

The first camera  $P$  can be arbitrarily chosen. A second camera  $M'$  can be obtained by composition of  $P$  and any transfer homography from the first view, as shown in Fig. 37 (left), with the necessary rank recovery procedure based on the epipole described in the previous section (Fig. 35).

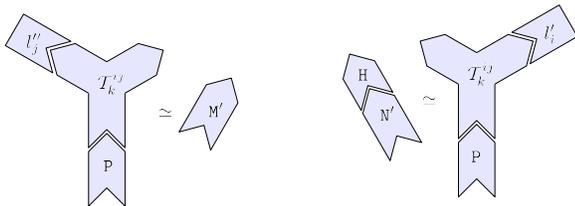


Figure 37. Extraction of compatible cameras from the Trifocal Tensor (the rank recovery step is not displayed).

The camera  $M'$  so obtained maps 3D points to the second view in a way which is compatible with  $\mathcal{T}$ . (This is of course equivalent to camera extraction from the fundamental matrix  $F_{12}$  extracted from  $\mathcal{T}$ .)

The third camera  $N'$  can be obtained in a similar way, but it must be carefully chosen to match the projective frame defined by the first two. This can be done by finding an appropriate correcting homography  $H$ . A particularly elegant

method is presented in [1], where transfers are induced by the epipoles, and  $H = e'e''^T - I$  is derived from algebraic considerations.

### 5.3. Four views

Purely algebraic methods for camera extraction from the Quadrifocal Tensor have been proposed by several authors [17, 18]. They are based on a reduced form of  $\mathcal{Q}$  with  $P_1 = [I|0]$  and the observation that some elements in the camera matrices can be deduced from the algebraic structure of the quadrifocal constraints.

This task is difficult for a purely diagrammatic approach because the structure of  $\mathcal{Q}$  has very little redundancy. In any case, compatible cameras can be extracted from  $\mathcal{Q}$  using algorithmic techniques described in the next section.

## 6. Diagrams as computational devices

Fig. 38 shows a circuit for simulation of  $\mathcal{T}$  from  $\mathcal{Q}$ . It obtains the line  $l$  induced by  $l'$  and  $l''$  as the join of the points induced by two different planes reprojected from the fourth view.

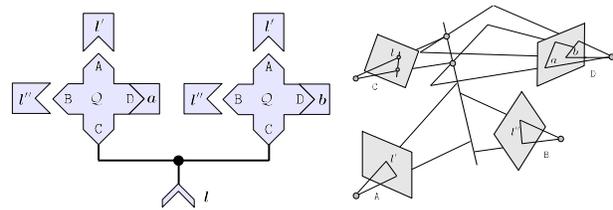


Figure 38. Simulation of  $\mathcal{T}$  from  $\mathcal{Q}$ .

Conversely, Fig. 39 shows a circuit to simulate  $\mathcal{Q}$  from a pair of Trifocal Tensors with two common cameras.

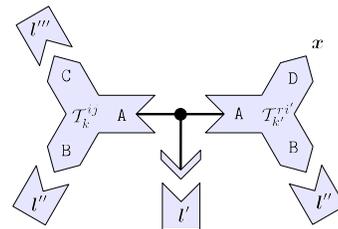


Figure 39. Simulation of  $\mathcal{Q}$  from two Trifocal Tensors.

(This beautiful construction admits the suggestive interpretation (Fig. 40) that one of the channels is simultaneously used as input ( $l'$ ) and output ( $AX$ )).

Some inputs must be connected to two different slots, so the above circuits are quadratic functions that cannot be collapsed into genuine tensors. In any case, we obtain effective algorithms which can be used for any desired purpose.

More importantly, diagrams automatically give straightforward implementations for the associated algorithms. For

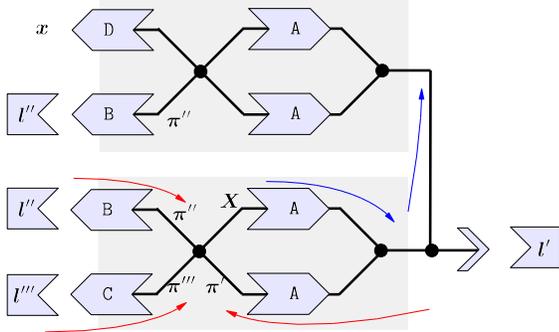


Figure 40. Geometric interpretation of data flow in Fig. 39.

example, the circuit for  $\mathcal{T}$  in Fig. 38 can be ultimately reduced to a simple 5-dimensional array with  $3^5 = 243$  entries. The computational engine is essentially based on contractions.

The tensor circuits described in this work have been checked using a freely available library for multilinear algebra [19]. The website contains updated material, including a tutorial and additional illustrative constructions.

## 7. Conclusion

We have developed diagrammatic tensor manipulation techniques for the specific requirements of Visual Geometry. The approach has been successfully applied to the analysis of several interesting situations. For example, we obtain circuits for the Fundamental Matrix and the Trifocal and Quadrifocal Tensors directly from geometrically meaningful building blocks. The diagrams expose the internal structure of the tensors, so they can be partially disassembled using mechanical graph reduction rules, providing effective procedures for camera extraction.

Tensor circuits also have practical computational advantages. They actually are direct implementations of the algorithms in terms of simple array contractions. Special linear algebra subroutines (for pseudoinversion, computation of null-spaces, etc.) are not required.

Diagram techniques must often be complemented with ordinary algebraic manipulation, but even in these cases the graphical approach is valuable, showing the steps in a derivation which arise from symmetries or redundant substructures.

In summary, the proposed diagrammatic approach is a powerful analysis tool for Visual Geometry applications. This technique can also be adapted to other Computer Vision fields based on multilinear algebra.

## Acknowledgements

This work has been supported by the Spanish MEC and European FEDER grants TIN2006-15516-C04-03 and Con-

solider CSD2006-00046. We thank the anonymous reviewers for their helpful comments.

## References

- [1] Richard Hartley and Andrew Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2 edition, 2003.
- [2] Olivier Faugeras, Quang Luong, and Theo Papadopoulos, *The Geometry of Multiple Images*, MIT Press, 2001.
- [3] C.T.J. Dodson and T. Poston, *Tensor Geometry*, Springer-Verlag, Graduate Texts in Mathematics 120, 1991.
- [4] Hermann Grassmann, *Die Lineale Ausdehnungslehre, ein neuer Zweig der Mathematik*, Leipzig, 1844, (Linear Extension Theory, a new branch of mathematics).
- [5] Desmond Fearnley-Sander, "Hermann Grassmann and the prehistory of universal algebra", *American Mathematical Monthly*, vol. 89, no. 3, pp. 161–166, 1982.
- [6] Anders Heyden, "Tensorial properties of multiple view constraints", *Mathematical Methods in the Applied Sciences*, vol. 23, no. 2, pp. 169–202, 2000.
- [7] S. Carlsson, "The double algebra: An effective tool for computing invariants in computer vision", *Lecture Notes in Computer Science*, vol. 825, pp. 145–164, 1994.
- [8] Bill Triggs, "The geometry of projective reconstruction i: Matching constraints and the joint image", in *Proceedings of the Int. Conf. on Computer Vision*, 1995, pp. 338–343.
- [9] Richard P. Feynman, "Space-time approach to nonrelativistic quantum mechanics", *Rev. Mod. Phys.* 20, 367, 1948.
- [10] Roger Penrose, *The Road to Reality*, Knopf, 2005.
- [11] Geoffrey E. Stedman, *Diagram Techniques in Group Theory*, Cambridge University Press, 1990.
- [12] Jim Blinn, "Uppers and downers: Part 2", *IEEE Computer Graphics and Applic.*, vol. 12, no. 3, pp. 80–85, 1992.
- [13] Bjorn K. Alsberg, "A diagram notation for n-mode array equations", *J. Chemometrics*, vol. 11, pp. 251–266, 1997.
- [14] Elisha Peterson, "A not-so-characteristic equation: the art of linear algebra", Dec 2007, arXiv:0712.2058v1.
- [15] Pedrag Cvitanovic, *Group Theory*, Princeton University Press, 2008.
- [16] Leo Dorst, Daniel Fontijne, and Stephen Mann, *Geometric Algebra for Computer Science: An Object-Oriented Approach to Geometry*, Morgan-Kaufmann, 2007.
- [17] Anders Heyden, *Geometry and Algebra of Multiple Projective Transformations*, PhD thesis, Department of Mathematics, Lund Institute of Technology, Sweden, 1995.
- [18] Richard Hartley, "Computation of the quadrifocal tensor", in *Proceedings of the 5th European Conference on Computer Vision*. 1998, pp. 20–35, Springer-Verlag.
- [19] "hTensor: A Haskell library for multilinear algebra.", 2009, <http://perception.inf.um.es/tensor>.