Trifocal tensor

In computer vision, the **trifocal tensor** (also **tritensor**) is a 3×3×3 array of numbers (i.e., a tensor) that incorporates all projective geometric relationships among three views. It relates the coordinates of corresponding points or lines in three views, being independent of the scene structure and depending only on the relative motion (i.e., pose) among the three views and their intrinsic calibration parameters. Hence, the trifocal tensor can be considered as the generalization of the fundamental matrix in three views. It is noted that despite the tensor being made up of 27 elements, only 18 of them are actually independent.

There is also a so-called **calibrated trifocal tensor**, which relates the coordinates of points and lines in three views given their intrinsic parameters and encodes the relative pose of the cameras up to global scale, totalling 11 independent elements or degrees of freedom. The reduced degrees of freedom allow for fewer correspondences to fit the model, at the cost of increased nonlinearity.[1]

Correlation slices

The tensor can also be seen as a collection of three rank-two 3 x 3 matrices $\mathbf{T_1}, \mathbf{T_2}, \mathbf{T_3}$ known as its *correlation slices*. Assuming that the projection matrices of three views are $\mathbf{P} = [\mathbf{I} \mid \mathbf{0}]$, $\mathbf{P'} = [\mathbf{A} \mid \mathbf{a_4}]$ and $\mathbf{P''} = [\mathbf{B} \mid \mathbf{b_4}]$, the correlation slices of the corresponding tensor can be expressed in closed form as $\mathbf{T_i} = \mathbf{a_i}\mathbf{b_4^t} - \mathbf{a_4}\mathbf{b_i^t}, i = 1\ldots3$, where $\mathbf{a_i}, \mathbf{b_i}$ are respectively the ith columns of the camera matrices. In practice, however, the tensor is estimated from point and line matches across the three views.

Trilinear constraints

One of the most important properties of the trifocal tensor is that it gives rise to linear relationships between lines and points in three images. More specifically, for triplets of corresponding points $\mathbf{x} \leftrightarrow \mathbf{x'} \leftrightarrow \mathbf{x''}$ and any corresponding lines $\mathbf{l} \leftrightarrow \mathbf{l'} \leftrightarrow \mathbf{l''}$ through them, the following *trilinear constraints* hold:

$$(\mathbf{l'^t} [\mathbf{T_1}, \mathbf{T_2}, \mathbf{T_3}] \mathbf{l''})[\mathbf{l}]_\times = \mathbf{0^t}$$

$$\mathbf{l'^t} \left(\sum_i x_i \mathbf{T_i}\right) \mathbf{l''} = 0$$

$$\mathbf{l'^t} \left(\sum_i x_i \mathbf{T_i}\right) [\mathbf{x''}]_\times = \mathbf{0^t}$$

$$[\mathbf{x'}]_\times \left(\sum_i x_i \mathbf{T_i}\right) \mathbf{l''} = \mathbf{0}$$

$$[\mathbf{x}']_\times \left(\sum_i x_i \mathbf{T}_i \right) [\mathbf{x}'']_\times = \mathbf{0}_{3\times3}$$

where $[\cdot]_\times$ denotes the skew-symmetric cross product matrix.

Transfer

Given the trifocal tensor of three views and a pair of matched points in two views, it is possible to determine the location of the point in the third view without any further information. This is known as *point transfer* and a similar result holds for lines and conics. For general curves, the transfer can be realized through a local differential curve model of osculating circles (i.e., curvature), which can then be transferred as conics.[2] The transfer of third-order models reflecting space torsion using calibrated trifocal tensors have been studied,[3] but remains an open problem for uncalibrated trifocal tensors.

Estimation

Uncalibrated

The classical case is 6 point correspondences[4][5] giving 3 solutions.

The case estimating the trifocal tensor from 9 line correspondences has only recently been solved.[6]

Calibrated

Estimating the calibrated trifocal tensor has been cited as notoriously difficult, and requires 4 point correspondences.[7]

The case of using only three point correspondences has recently been solved, where the points are attributed with tangent directions or incident lines; with only two of the points having incident lines, this is a minimal problem of degree 312 (so there can be at most 312 solutions) and is relevant for the case of general curves (whose points have tangents), or feature points with attributed directions (such as SIFT directions).[8] The same technique solved the mixed case of three point correspondences and one line correspondence, which has also been shown to be minimal with degree 216.

References

1. Martyushev, E. V. (2017). "On Some Properties of Calibrated Trifocal Tensors". *Journal of Mathematical Imaging and Vision*. **58** (2): 321–332. arXiv:1601.01467 (https://arxiv.org/abs/1601.01467) . doi:10.1007/s10851-017-0712-x (https://doi.org/10.1007%2Fs10851-017-0712-x) . S2CID 1634602 (https://api.semanticscholar.org/CorpusID:1634602) .

2. Schmid, Cordelia (2000). "The Geometry and Matching of Lines and Curves Over Multiple Views" (http://hal.archives-ouvertes.fr/docs/00/54/83/01/PDF/ijcv.pdf) (PDF). *International Journal of Computer Vision*. **40** (3): 199–233. doi:10.1023/A:1008135310502 (https://doi.org/10.1023%2FA%3A1008135310502) . S2CID 11844321 (https://api.semanticscholar.org/CorpusID:11844321) .

3. Fabbri, Ricardo; Kimia, Benjamin (2016). "Multiview Differential Geometry of Curves". *International Journal of Computer Vision*. **120** (3): 324–346. arXiv:1604.08256 (https://arxiv.org/abs/1604.08256) . Bibcode:2016arXiv160408256F (https://ui.adsabs.harvard.edu/abs/2016arXiv160408256F) . doi:10.1007/s11263-016-0912-7 (https://doi.org/10.1007%2Fs11263-016-0912-7) . S2CID 11908870 (https://api.semanticscholar.org/CorpusID:11908870) .

4. Richard Hartley and Andrew Zisserman (2003). "Online Chapter: Trifocal Tensor" (http://www.robots.ox.ac.uk/~vgg/hzbook/hzbook2/HZtrifocal.pdf) (PDF). *Multiple View Geometry in computer vision*. Cambridge University Press. ISBN 978-0-521-54051-3.

5. Heyden, A. (1995). "Reconstruction from Image Sequences by means of Relative Depths". *Proceedings of IEEE International Conference on Computer Vision*. pp. 1058–1063. doi:10.1109/ICCV.1995.466817 (https://doi.org/10.1109%2FICCV.1995.466817) . ISBN 0-8186-7042-8. S2CID 7789642 (https://api.semanticscholar.org/CorpusID:7789642) .

6. Larsson, Viktor; Astrom, Kalle; Oskarsson, Magnus (2017). "Efficient Solvers for Minimal Problems by Syzygy-Based Reduction" (http://lup.lub.lu.se/record/e10f7381-3eeb-43e8-b138-5c3f8c2cde74) . *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 2383–2392. doi:10.1109/CVPR.2017.256 (https://doi.org/10.1109%2FCVPR.2017.256) . ISBN 978-1-5386-0457-1. S2CID 13069612 (https://api.semanticscholar.org/CorpusID:13069612) .

7. Nister, David; Schaffalitzky, Frederik (2006). "Four Points in Two or Three Calibrated Views: Theory and Practice". *International Journal of Computer Vision*. **67** (2): 211–231. doi:10.1007/s11263-005-4265-x (https://doi.org/10.1007%2Fs11263-005-4265-x) . S2CID 10231211 (https://api.semanticscholar.org/CorpusID:10231211) .

8. Fabbri, Ricardo; Duff, Timothy; Fan, Hongyi; Regan, Margaret; de Pinho, David; Tsigaridas, Elias; Wampler, Charles; Hauenstein, Jonathan; Kimia, Benjamin; Leykin, Anton; Pajdla, Tomas (23 Mar 2019). "Trifocal Relative Pose from Lines at Points and its Efficient Solution".

arXiv:1903.09755 (https://arxiv.org/abs/1903.09755) [cs.CV (https://arxiv.org/archive/cs.CV)].

Further reading

- Hartley, Richard I. (1997). "Lines and Points in Three Views and the Trifocal Tensor". *International Journal of Computer Vision*. **22** (2): 125–140. doi:10.1023/A:1007936012022 (https://doi.org/10.1023%2FA%3A1007936012022) . S2CID 8979544 (https://api.semanticscholar.org/CorpusID:8979544) .

- Torr, P. H. S.; Zisserman, A. (1997). "Robust Parameterization and Computation of the Trifocal Tensor". *Image and Vision Computing*. **15** (8): 591–607. CiteSeerX 10.1.1.41.3172 (https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.41.3172) . doi:10.1016/S0262-8856(97)00010-3 (https://doi.org/10.1016%2FS0262-8856%2897%2900010-3) .

External links

- Visualization of trifocal geometry (https://web.archive.org/web/20091120063117/http://www2.informatik.hu-berlin.de/~blaschek/diplvortrag/learn_epi/EpipolarGeo.html) (originally by Sylvain Bougnoux of INRIA Robotvis, requires Java)

Algorithms

- Matlab implementation of the uncalibrated trifocal tensor (https://github.com/LauraFJulia/TFT_vs_Fund) estimation and comparison to pairwise fundamental matrices

- C++ implementation of the calibrated trifocal tensor (https://github.com/rfabbri/minus) estimation using optimized Homotopy Continuation code. Presently includes cases of three corresponding points with lines at these points (as in feature positions and orientations, or curve points with tangents), and also for three corresponding points and one line correspondence.