A Walkthrough on V-SLAM & VO

Part I – Localisation

Chia-Yen Chen

Content and videos courtesy of

Hsiang-Jen (Johnny) Chien

Centre of Robotics Vision

School of Engineering, Computer & Mathematical Sciences, AUT

Background

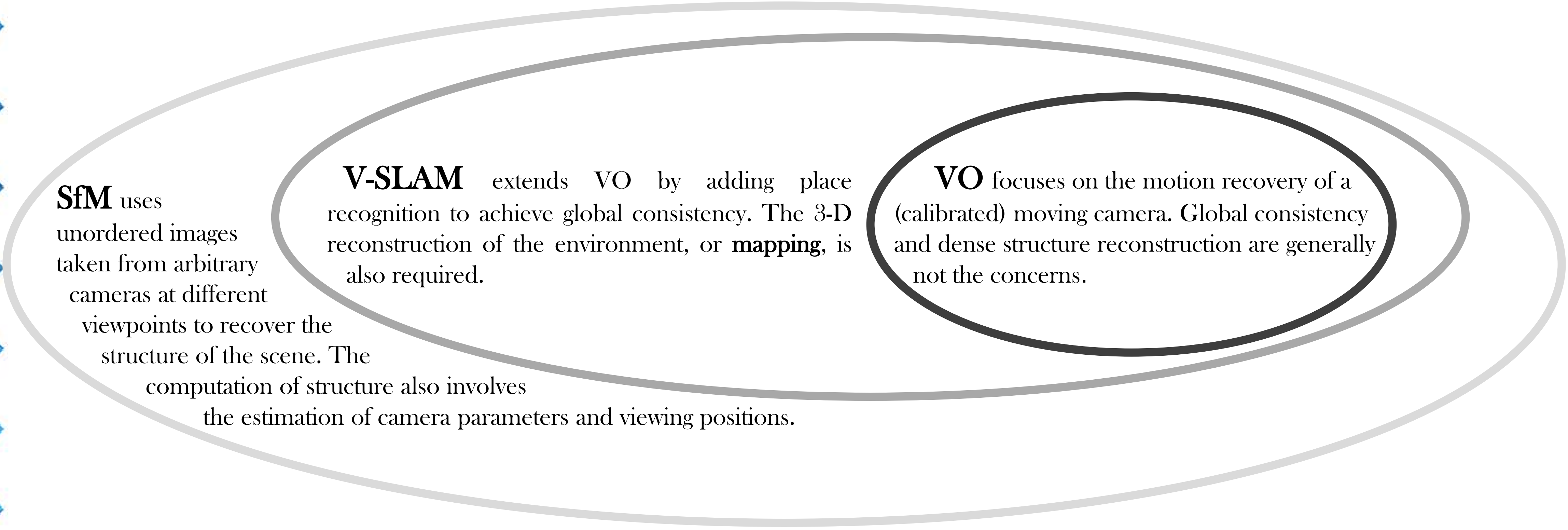
Tracking

Mapping

Summary

SfM, V-SLAM and VO

- Structure from Motion, Visual Simultaneous Localisation and Mapping, and Visual Odometry are closely related topics



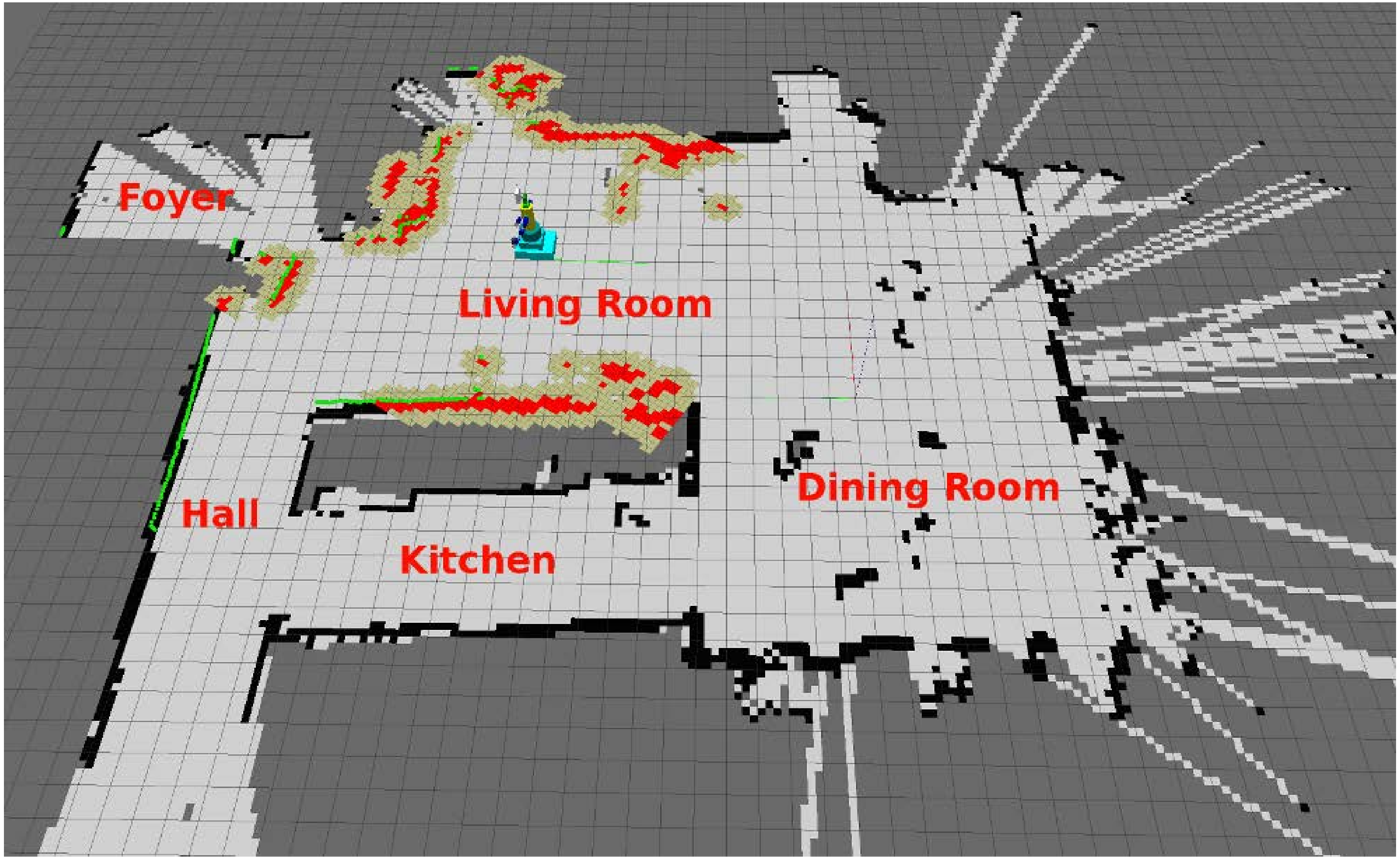
Structure from Motion

photogrammetry

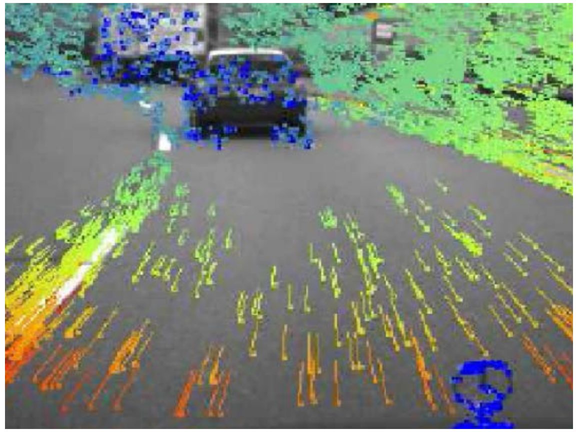
- Microsoft Photosynth https://photosynth.net/ (closing..)
- VisualSFM http://ccwu.me/vsfm/install.html
- Bundler https://www.cs.cornell.edu/~snavely/bundler/
- OpenMVG http://openmvg.readthedocs.io/

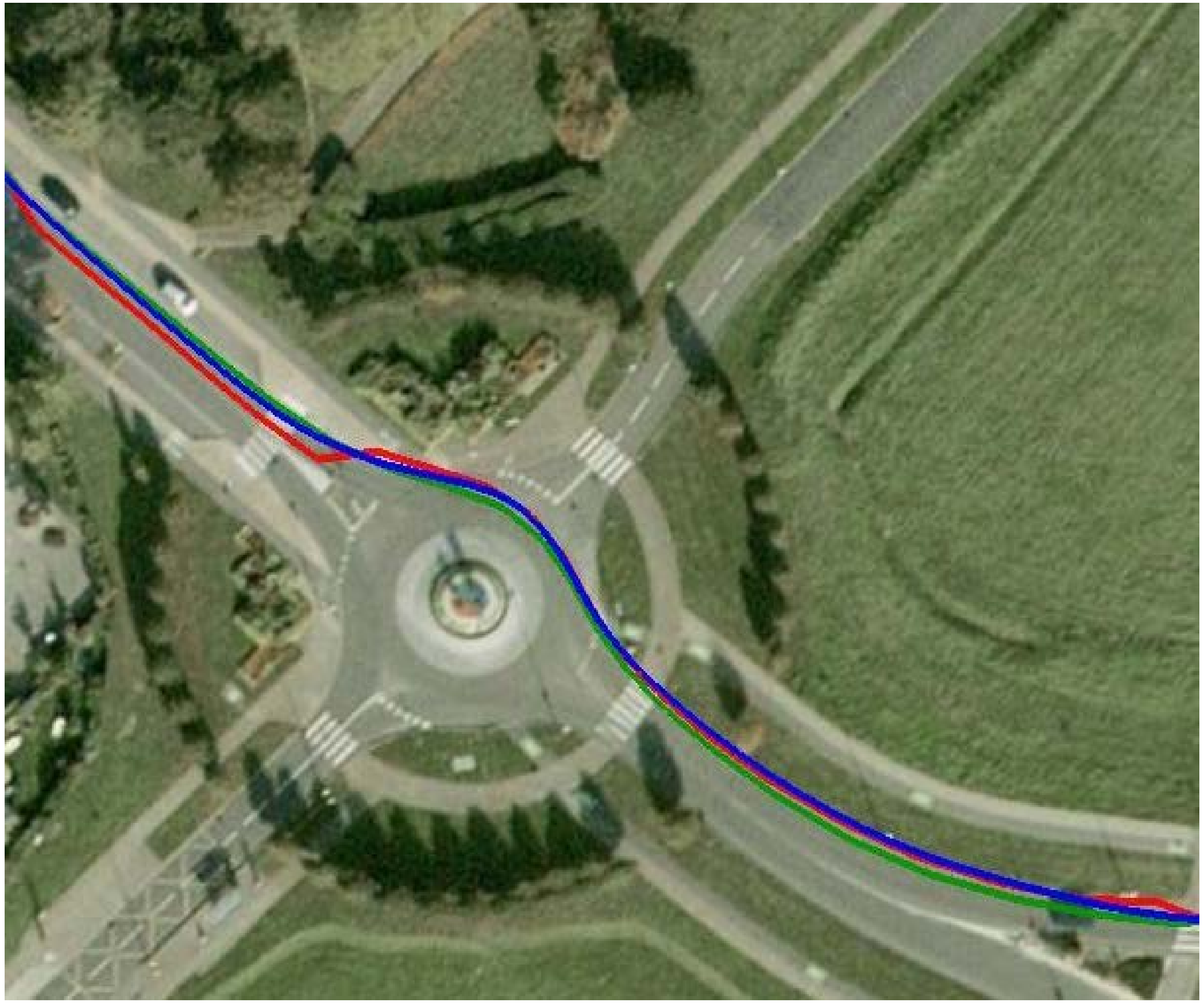
V-SLAM

robotics / machine vision

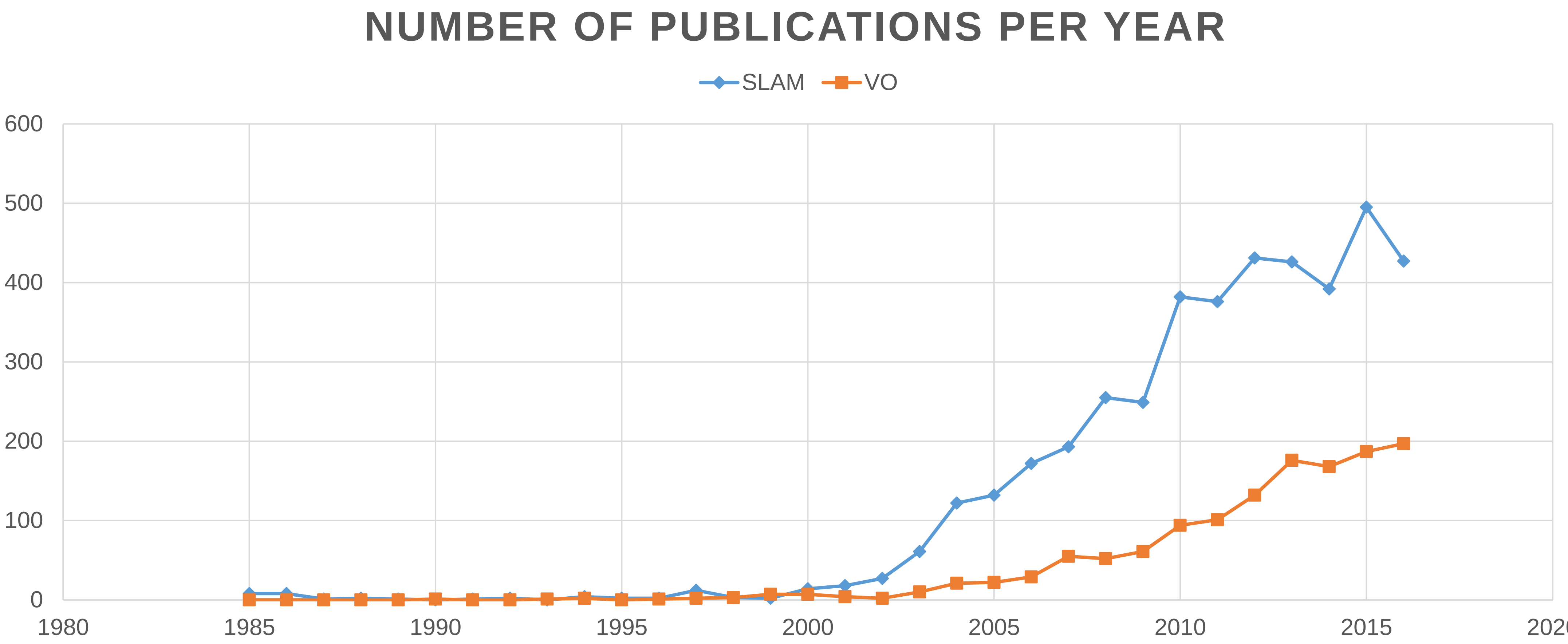


Visual Odometry



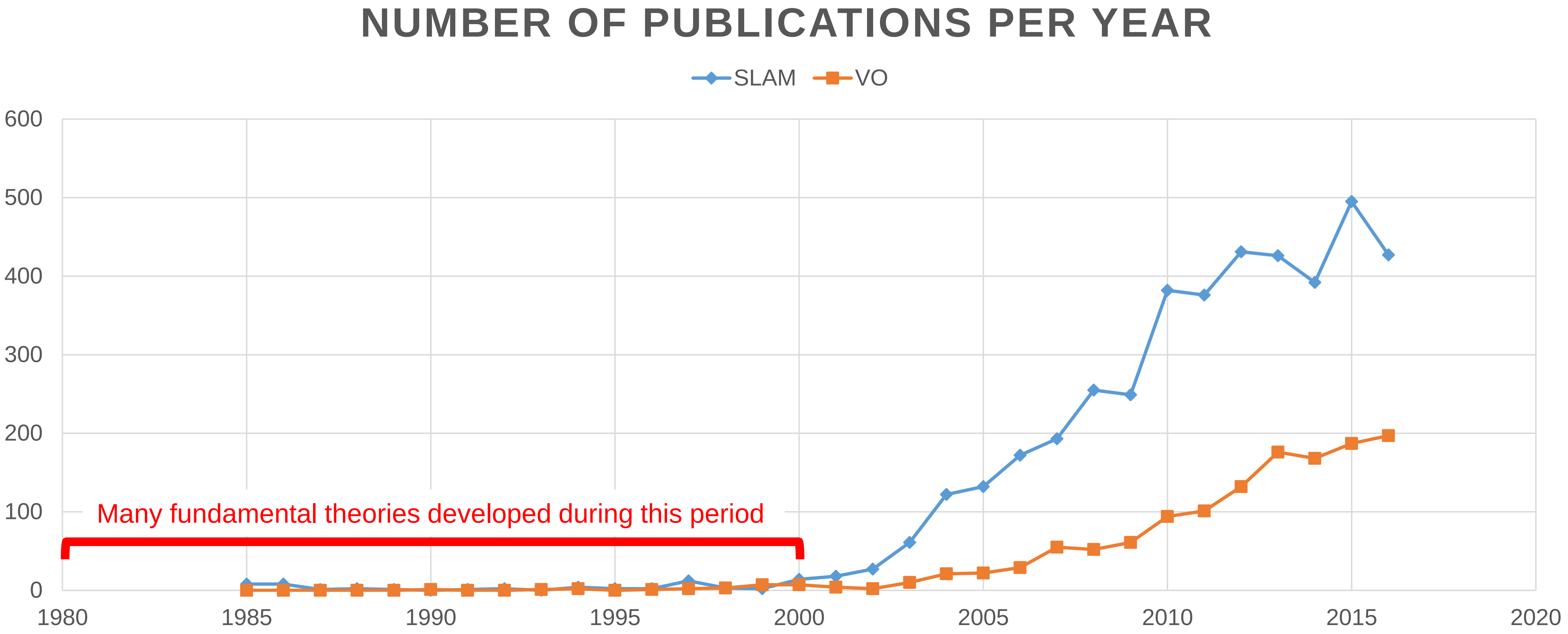


Trend

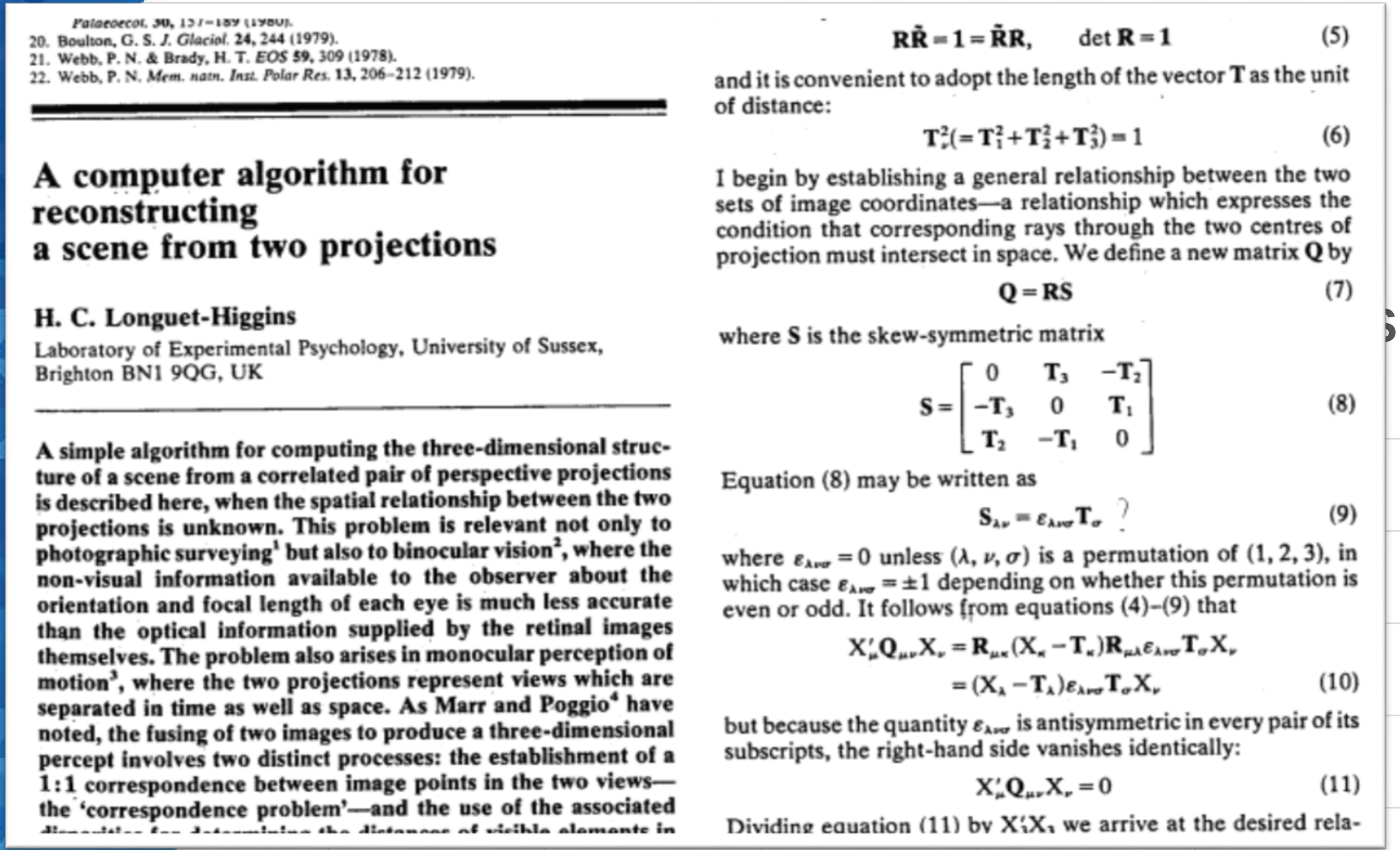


Source: ELSEVIER SCOPUS

Trend



Source: ELSEVIER SCOPUS



Palaeoecol. **30,** 157–189 (1980).
20. Boulton, G. S. *J. Glaciol.* **24,** 244 (1979).
21. Webb, P. N. & Brady, H. T. *EOS* **59,** 309 (1978).
22. Webb, P. N. *Mem. natn. Inst. Polar Res.* **13,** 206–212 (1979).

A computer algorithm for reconstructing a scene from two projections

H. C. Longuet-Higgins

Laboratory of Experimental Psychology, University of Sussex, Brighton BN1 9QG, UK

A simple algorithm for computing the three-dimensional structure of a scene from a correlated pair of perspective projections is described here, when the spatial relationship between the two projections is unknown. This problem is relevant not only to photographic surveying[1] but also to binocular vision[2], where the non-visual information available to the observer about the orientation and focal length of each eye is much less accurate than the optical information supplied by the retinal images themselves. The problem also arises in monocular perception of motion[3], where the two projections represent views which are separated in time as well as space. As Marr and Poggio[4] have noted, the fusing of two images to produce a three-dimensional percept involves two distinct processes: the establishment of a 1:1 correspondence between image points in the two views—the 'correspondence problem'—and the use of the associated disparities for determining the distances of visible elements in

$$\mathbf{R}\tilde{\mathbf{R}} = 1 = \tilde{\mathbf{R}}\mathbf{R}, \qquad \det \mathbf{R} = 1 \tag{5}$$

and it is convenient to adopt the length of the vector **T** as the unit of distance:

$$\mathbf{T}_\nu^2 (= \mathbf{T}_1^2 + \mathbf{T}_2^2 + \mathbf{T}_3^2) = 1 \tag{6}$$

I begin by establishing a general relationship between the two sets of image coordinates—a relationship which expresses the condition that corresponding rays through the two centres of projection must intersect in space. We define a new matrix **Q** by

$$\mathbf{Q} = \mathbf{RS} \tag{7}$$

where **S** is the skew-symmetric matrix

$$\mathbf{S} = \begin{bmatrix} 0 & \mathbf{T}_3 & -\mathbf{T}_2 \\ -\mathbf{T}_3 & 0 & \mathbf{T}_1 \\ \mathbf{T}_2 & -\mathbf{T}_1 & 0 \end{bmatrix} \tag{8}$$

Equation (8) may be written as

$$\mathbf{S}_{\lambda\nu} = \varepsilon_{\lambda\nu\sigma}\mathbf{T}_\sigma \tag{9}$$

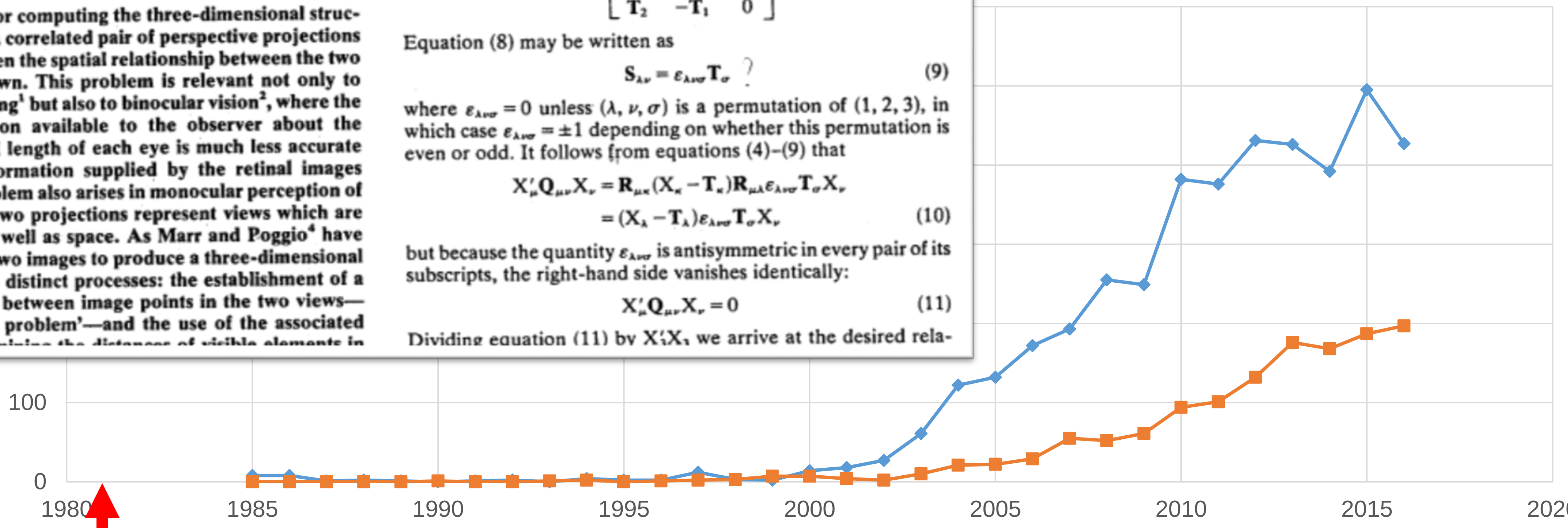
where $\varepsilon_{\lambda\nu\sigma} = 0$ unless (λ, ν, σ) is a permutation of (1, 2, 3), in which case $\varepsilon_{\lambda\nu\sigma} = \pm 1$ depending on whether this permutation is even or odd. It follows from equations (4)–(9) that

$$\begin{aligned} \mathbf{X}'_\mu \mathbf{Q}_{\mu\nu}\mathbf{X}_\nu &= \mathbf{R}_{\mu\kappa}(\mathbf{X}_\kappa - \mathbf{T}_\kappa)\mathbf{R}_{\mu\lambda}\varepsilon_{\lambda\nu\sigma}\mathbf{T}_\sigma\mathbf{X}_\nu \\ &= (\mathbf{X}_\lambda - \mathbf{T}_\lambda)\varepsilon_{\lambda\nu\sigma}\mathbf{T}_\sigma\mathbf{X}_\nu \end{aligned} \tag{10}$$

but because the quantity $\varepsilon_{\lambda\nu\sigma}$ is antisymmetric in every pair of its subscripts, the right-hand side vanishes identically:

$$\mathbf{X}'_\mu \mathbf{Q}_{\mu\nu}\mathbf{X}_\nu = 0 \tag{11}$$

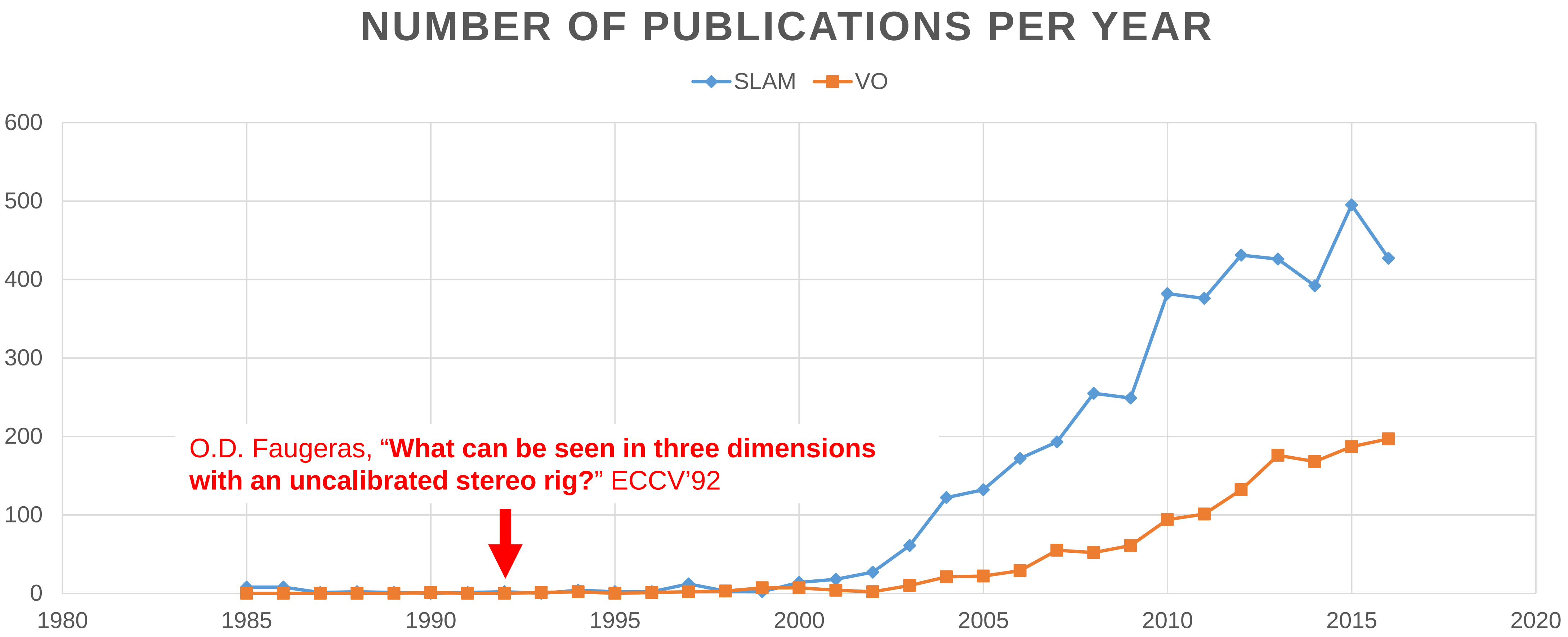
Dividing equation (11) by $\mathbf{X}'_3\mathbf{X}_3$ we arrive at the desired rela-



H. C. Longuet-Higgins, "**A computer algorithm for reconstructing a scene from two projections.**" Nature, vol. 291, pp. 133-135, Sep 1981.

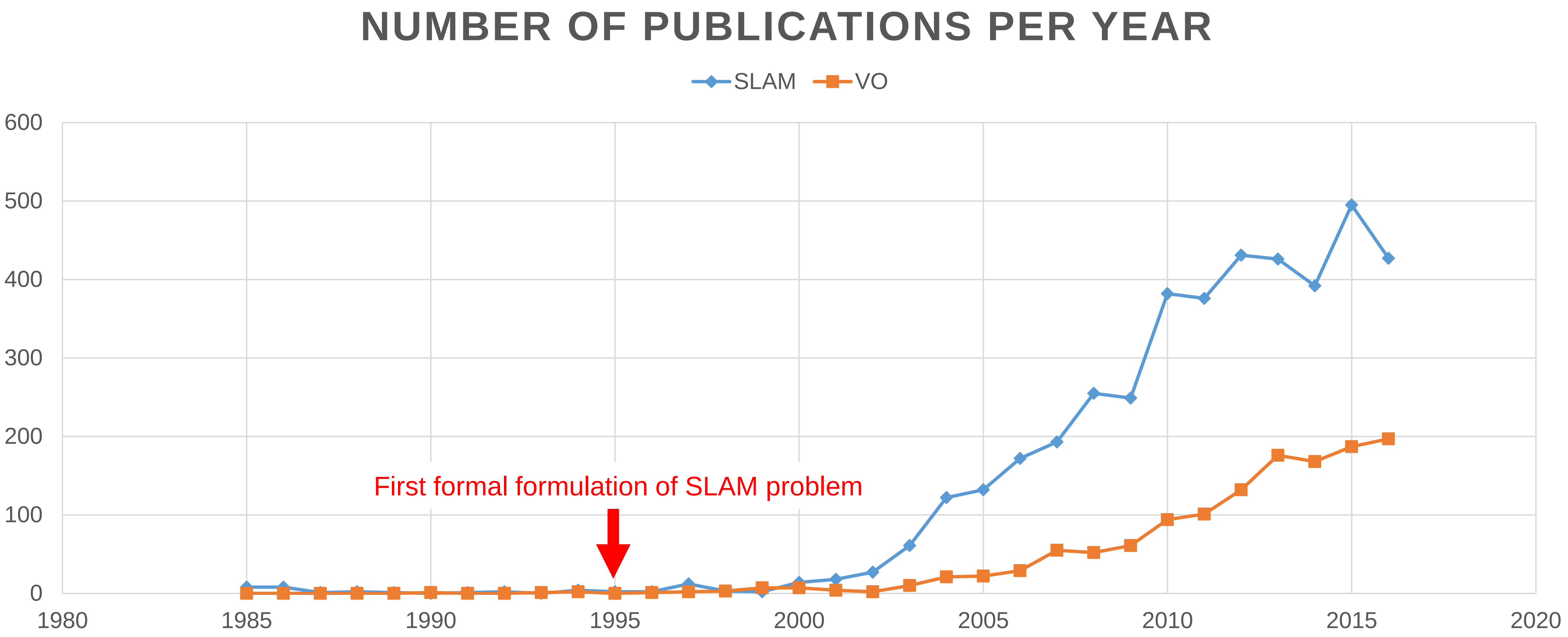
Source: ELSEVIER SCOPUS

Trend



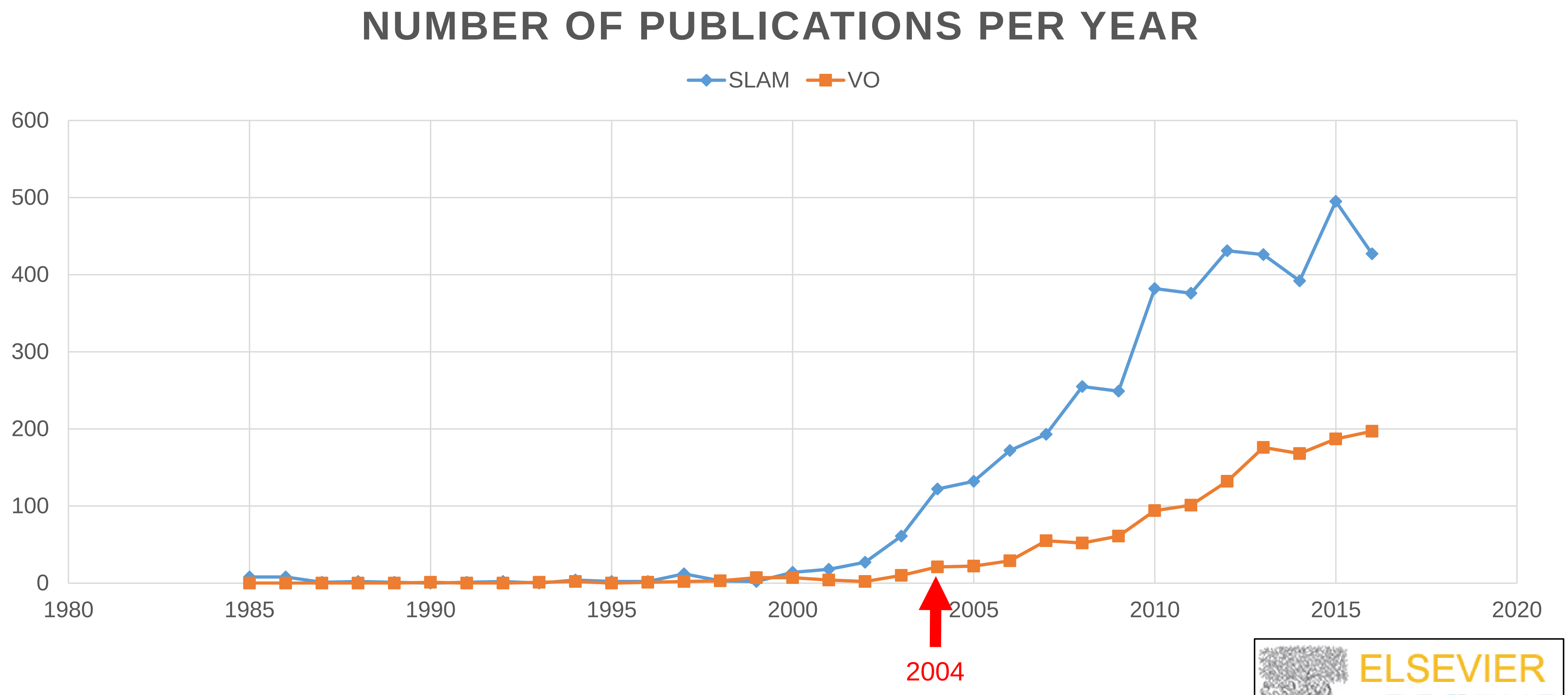
Source: ELSEVIER SCOPUS

Trend

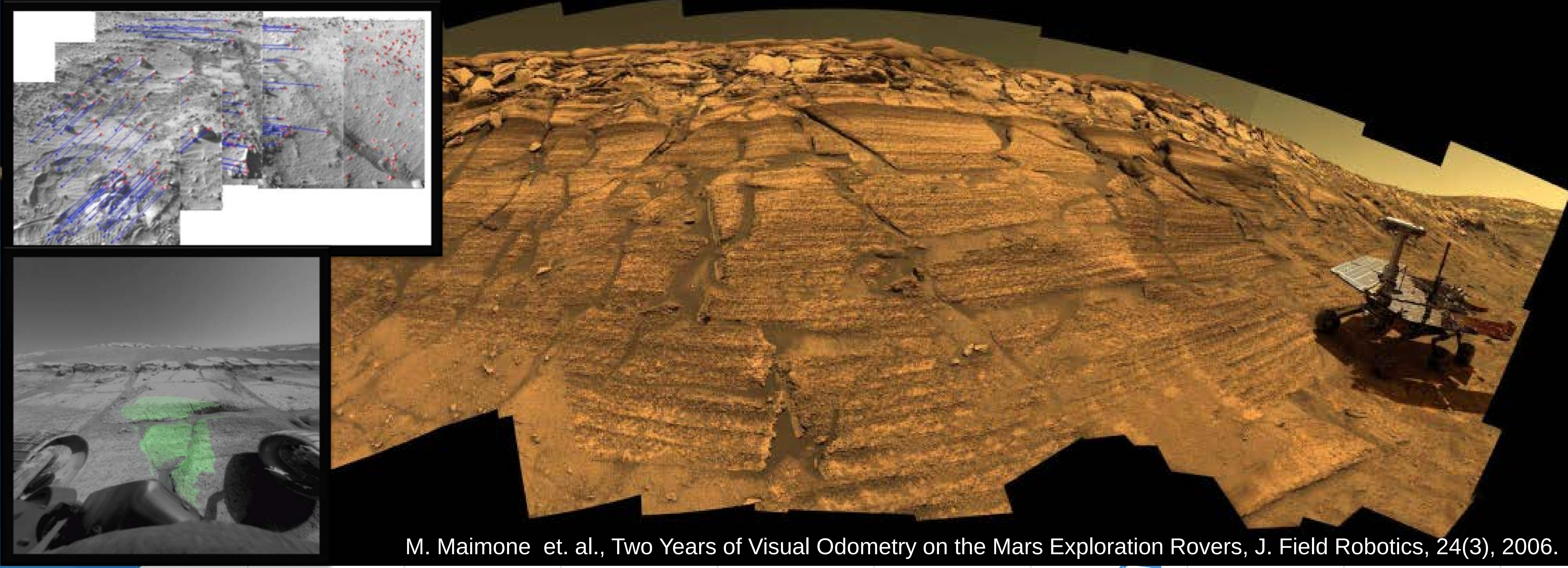


Source: ELSEVIER SCOPUS

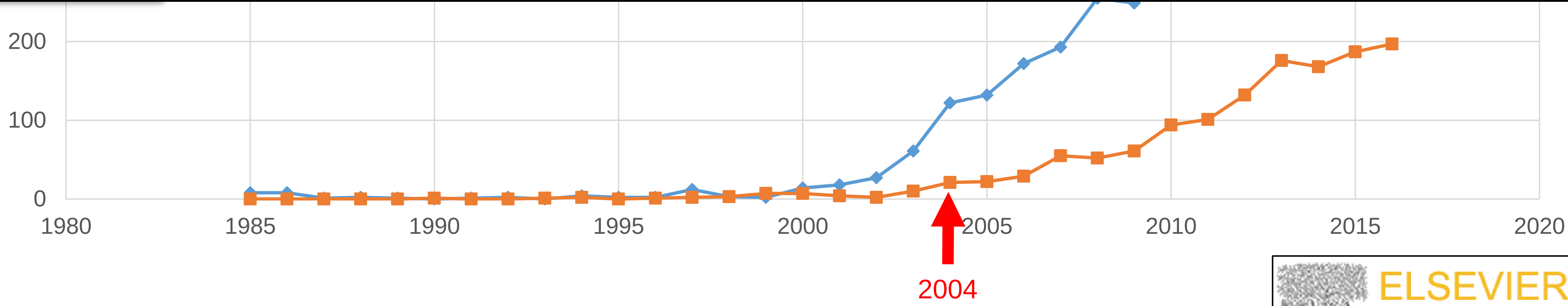
Trend



Source: ELSEVIER SCOPUS

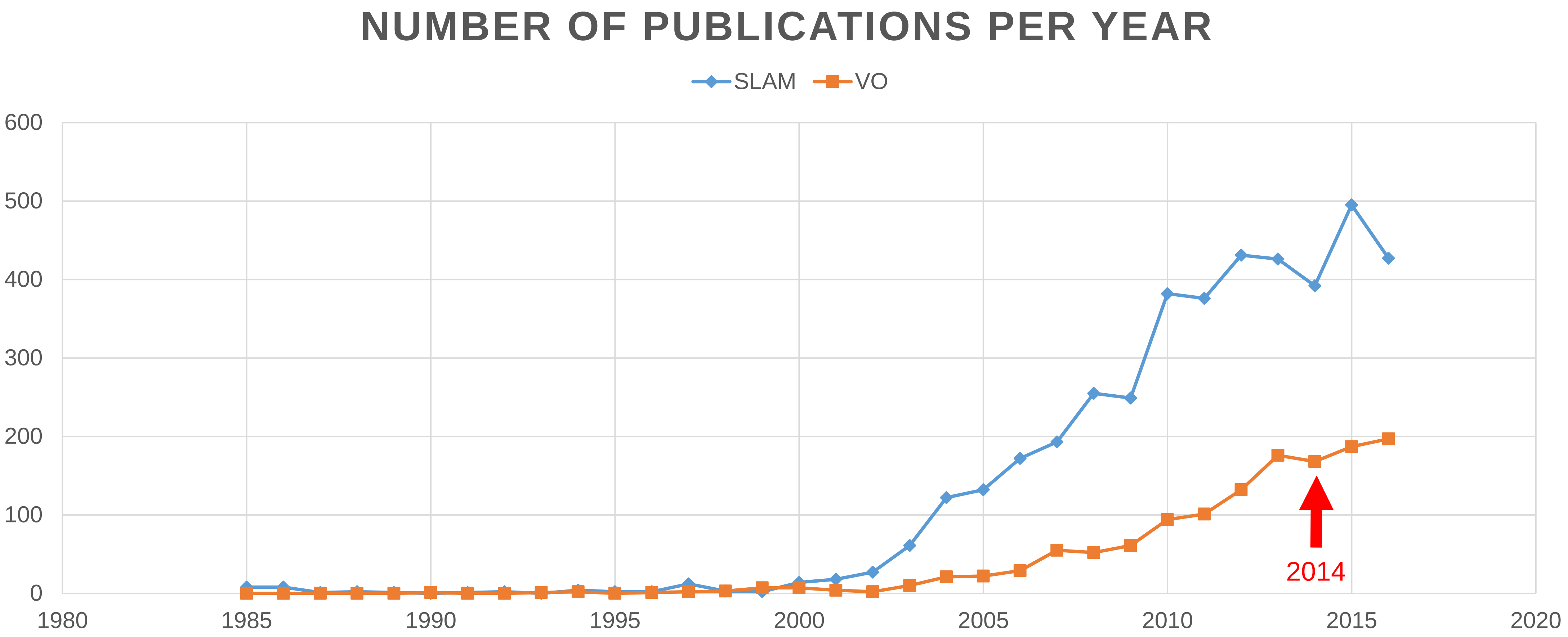


M. Maimone et. al., Two Years of Visual Odometry on the Mars Exploration Rovers, J. Field Robotics, 24(3), 2006.

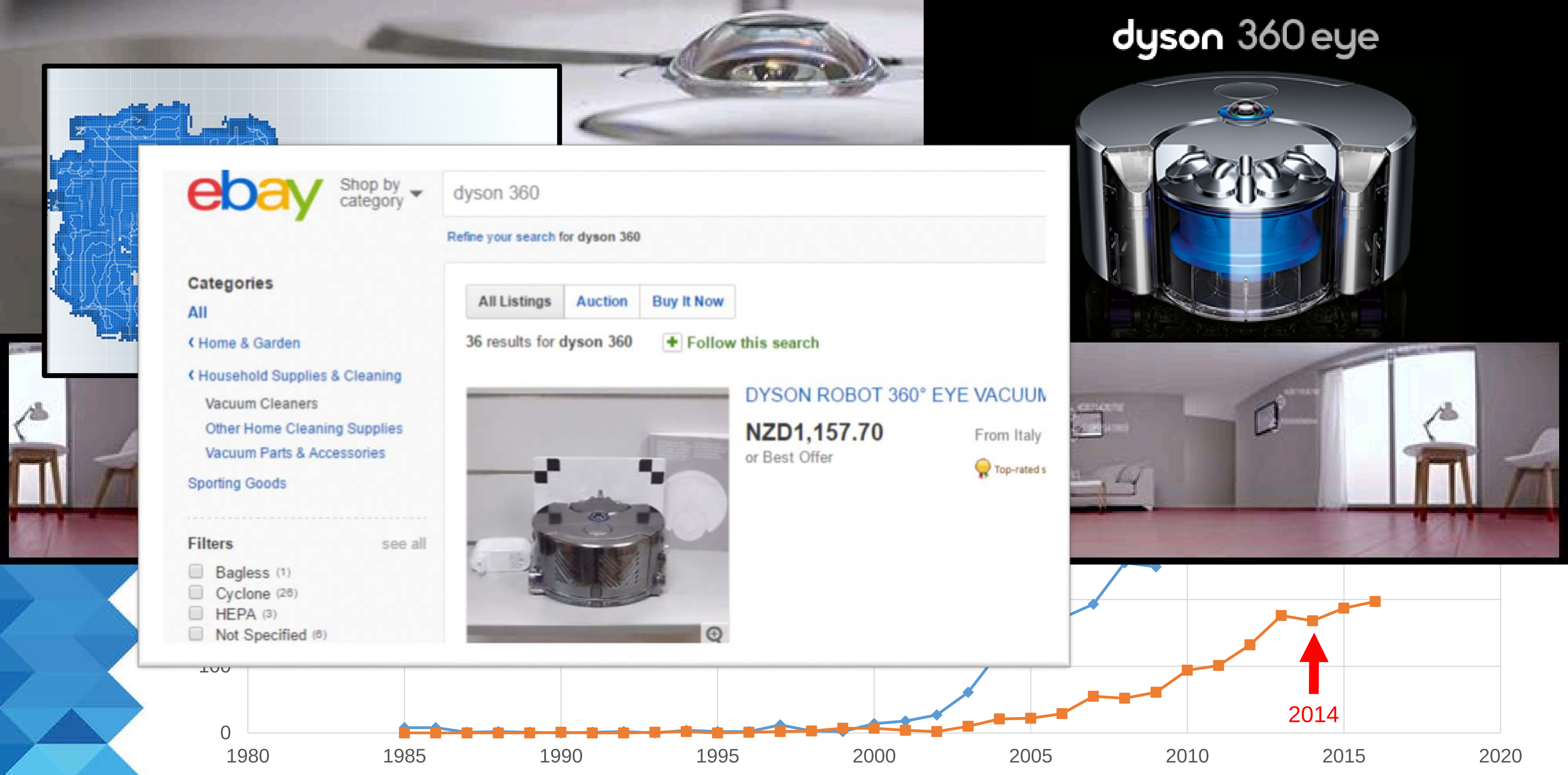


Source:

Trend

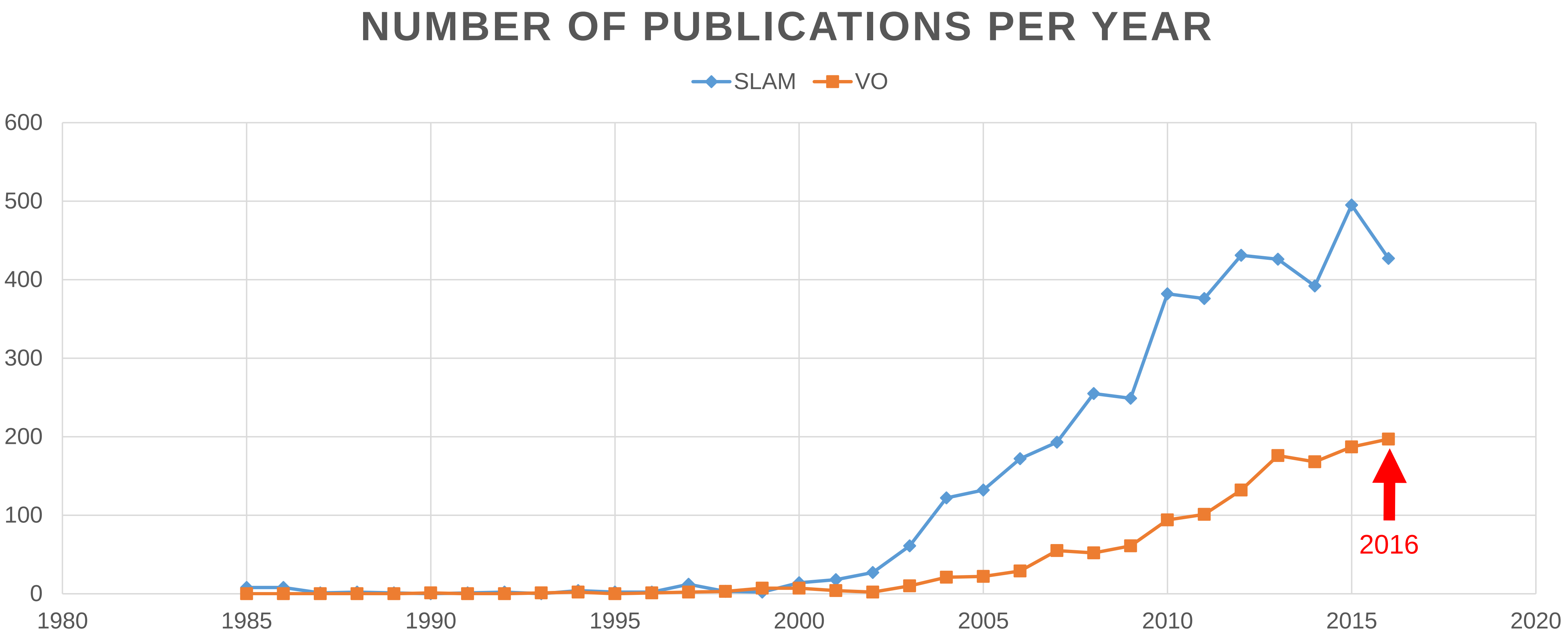


Source: ELSEVIER SCOPUS



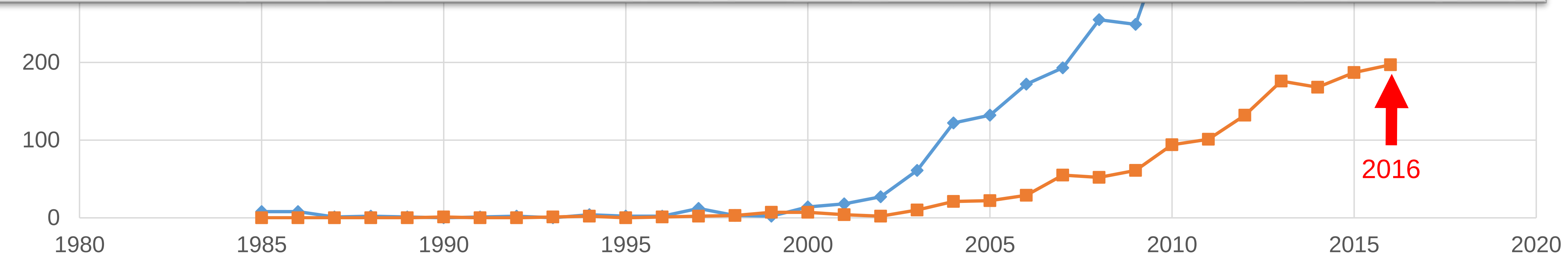
Source: ELSEVIER SCOPUS

Trend



Source: ELSEVIER SCOPUS

	Document title	Authors	Year	Source
2	Adaptive feature tracking with kalman filter for ego-motion estimation articlelinker View at Publisher	Huang, T.-H., Chuang, C.-C., Chang, Y.-H., Chen, C.-Y.	2016	Proceedings - 2016 IEEE 2nd International Conference on Multimedia Big Data, BigMM 2016
3	Multi-frame feature integration for multi-camera visual odometry articlelinker View at Publisher	Chien, H.-J., Geng, H., Chen, C.-Y., Klette, R.	2016	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)
4	Visual odometry in dynamic environments with geometric multi-layer optimisation articlelinker | View at Publisher | 顯示摘要 | 相關文獻	Geng, H., Chien, H.-J., Nicolescu, R., Klette, R.	2016	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 9992 LNAI, pp. 183-190
5	Multi-run: An approach for filling in missing information of 3D roadside reconstruction	Geng, H., Chien, H.-J., Klette, R.	2016	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)



Source: ELSEVIER SCOPUS

Configurations

Monocular Vision

Binocular Vision

Trinocular Vision

Monocular + LiDAR

Stereo + LiDAR

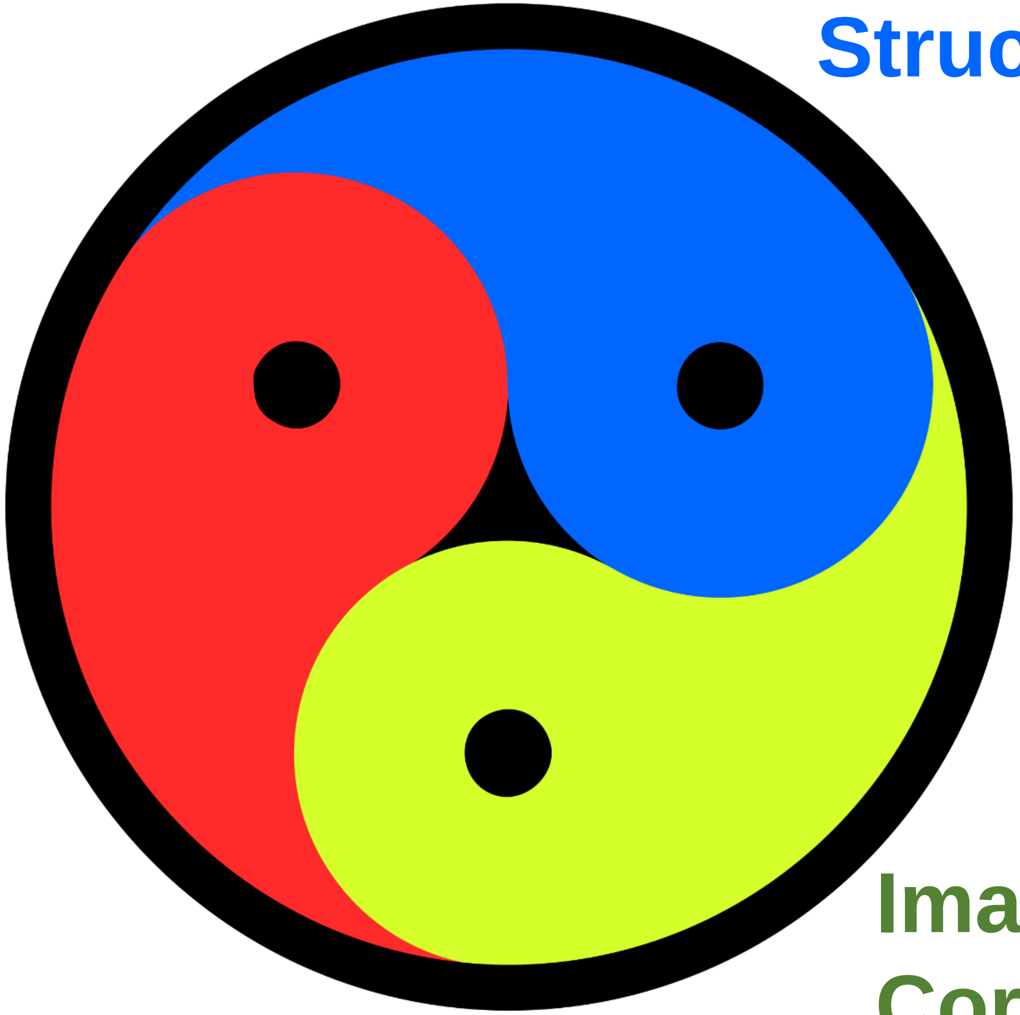
RGB-D Camera

RGB-D + Stereo

The Triality

3D Scene Structure

Camera Motion



given these

find this

Image Correspondences

What a SLAM system looks Like..

New Frame
(with 3D structure)

Feature Extraction & Tracking
(to establish image correspondences)

Ego-motion Estimation
(to recover camera motion)

Visual Odometry
(Localisation / Tracking)

Key-frame Selection

Loop Detection

Map Update / Bundle Adjustment

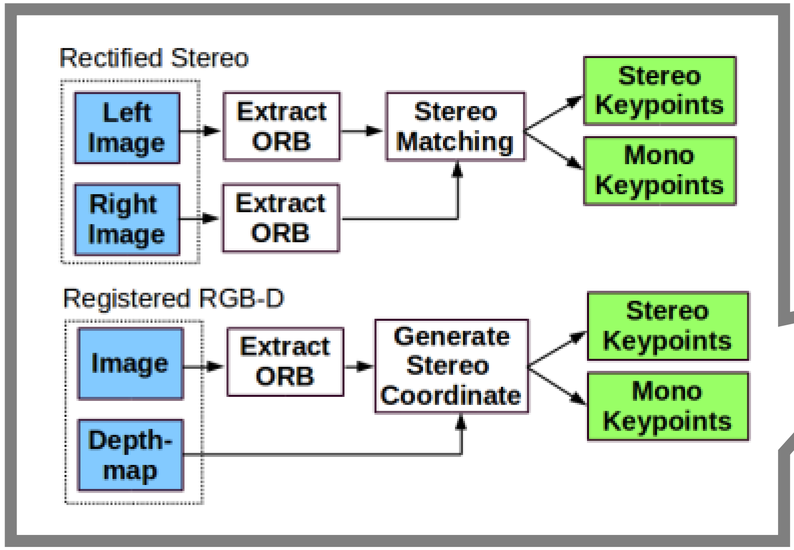
Mapping

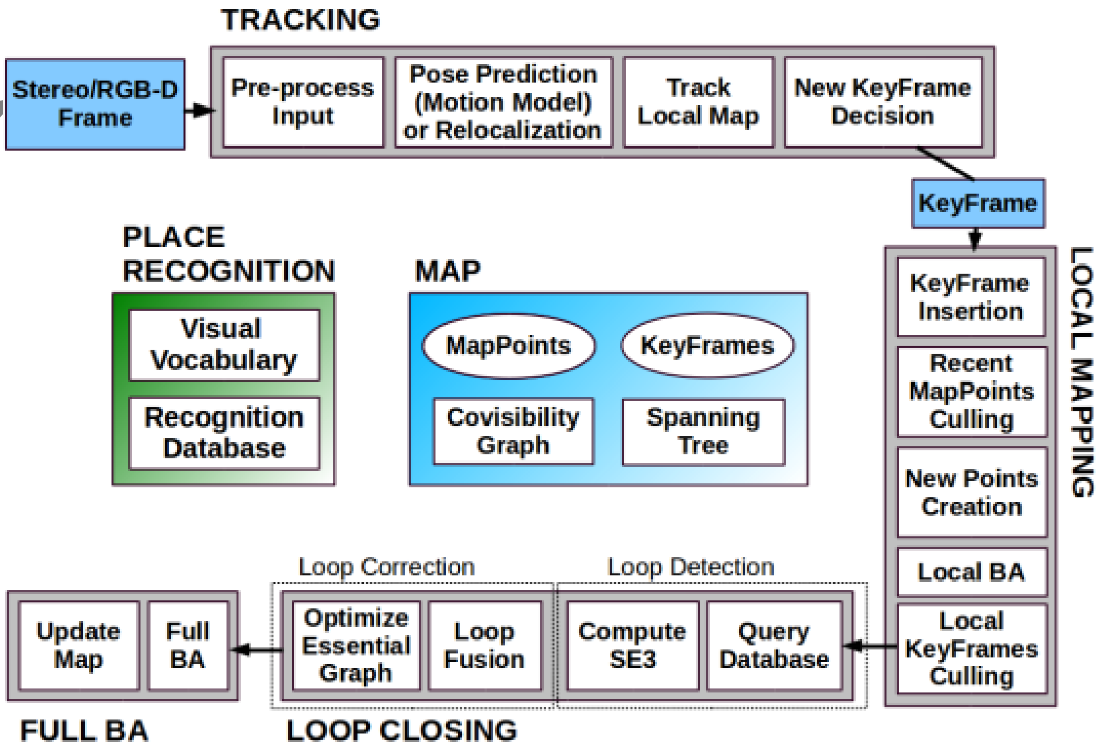
ORB-SLAM 1 & 2

Raúl Mur-Artal et. al., Universidad Zaragoza, Spain, 2015-16

190 citations

- Very fast feature detection and extraction (ORB = FAST + Oriented BRIEF)





- Key frames
- Loop closure
- Bundle Adjustment

Background

Tracking

Mapping

Summary

Tracking

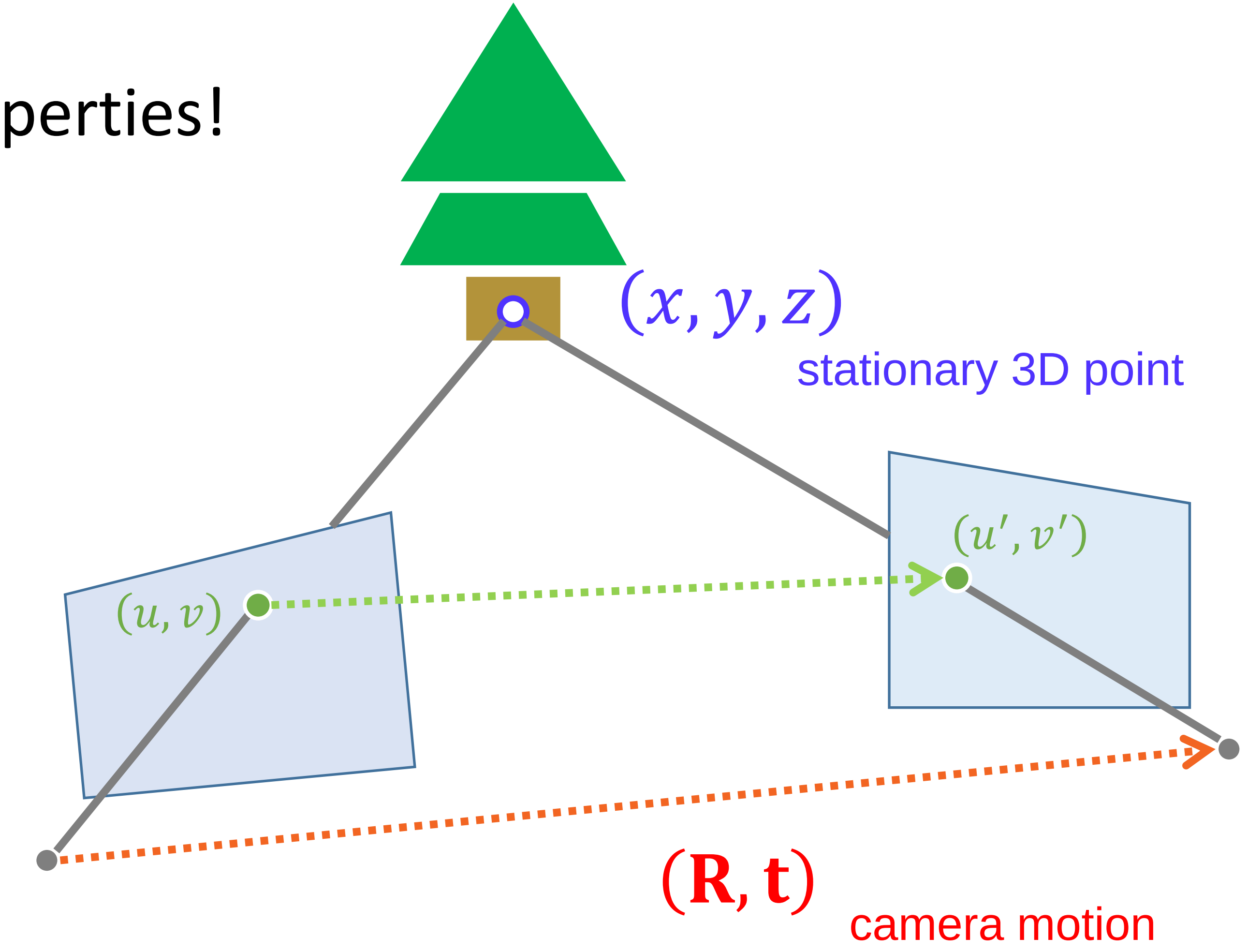
- To continuously solve the system's ego-motion from each two consequent frames
- The motion is modelled by a 3D Euclidean transform
  - …which can be represented by a rotation matrix and a translation vector (i.e. 6-dof)
- The current position of system is determined by concatenating a series of transforms
- Known as *dead reckoning* in terms of navigation
  - "dead" derived from deduced, or ded

How can a rigid transform be derived from two images?

- Use motion-invariant properties!

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \sim \mathbf{K} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} \sim \mathbf{K} \left(\mathbf{R} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \mathbf{t} \right)$$



Two branches

- Indirect Methods (feature-based)
  - Transform image pixels to a carefully crafted **feature space**
  - Matching is performed in the feature space, before ego-motion estimation
  - Usually sparse key points are picked
  - Faster and dominating VO/SLAM for decades
- Direct Methods (feature-free)
  - Use **pixel intensities** directly
  - Matching simultaneously happens when solving ego-motion
  - Could be dense, semi-dense or sparse
  - Slow but becoming popular due to advances in parallel computing

Problem formulation

- A feature-based method finds the motion that minimises geodesic distances of the corresponding pixels

Note: the correspondence $g_i \rightarrow \rho_i'$ is known

$$\Phi_{geo}(\mathbf{R},\mathbf{t}) = \sum_{1\le i\le n} \|\boldsymbol{\rho}_i' - \pi(\mathbf{R}\mathbf{g}_i + \mathbf{t})\|^2$$

camera motion

corresponding pixel coordinates in frame $k+1$

feature's 3D coordinates in frame k

- While a direct method finds the motion that minimises photometric differences without knowing pixel correspondences

No image correspondence!

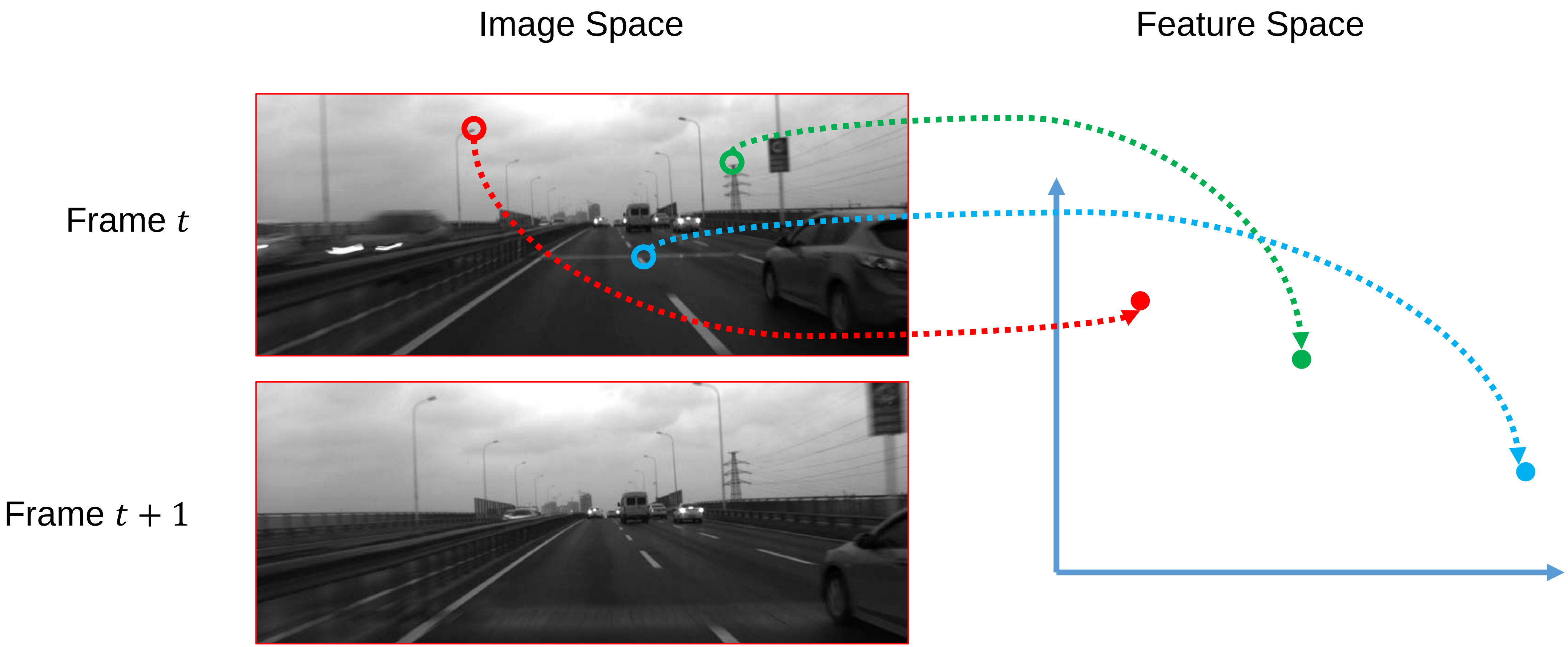
$$\Phi_{photo}(\mathbf{R},\mathbf{t}) = \sum_{1\le i\le n} \left\|\mathbf{I}(\boldsymbol{\rho}_i) - \mathbf{I}'\left(\pi(\mathbf{R}\mathbf{g}_i + \mathbf{t})\right)\right\|^2$$

camera motion

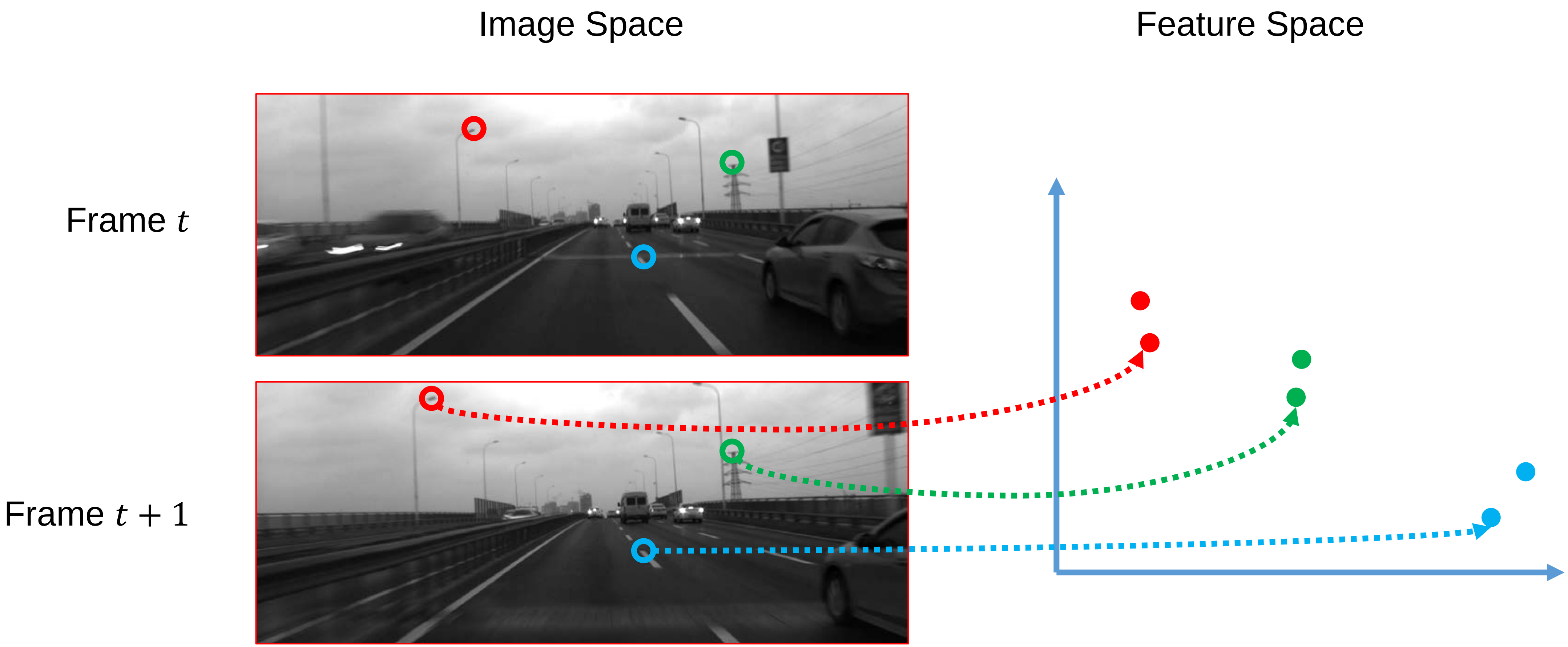
pixel coordinates in frame k

pixel's 3D coordinates in frame k

Indirect method



Indirect method

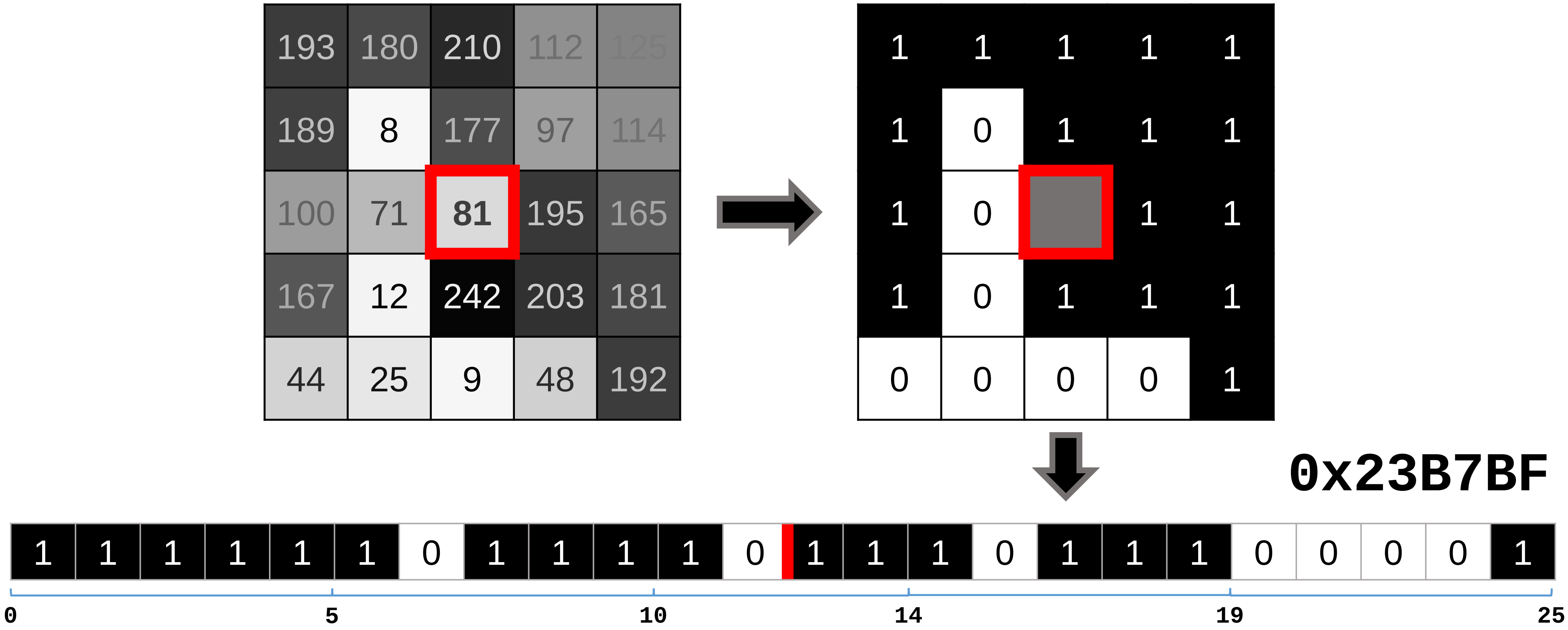


Indirect method

Census transform

An example of feature space

- Encodes local intensity pattern



Feature matching

- Given 2 sets of features $\mathcal{F}, \mathcal{F}'$ and ν a feature space transform function
- for each $\chi \in \mathcal{F}$ we find a $\chi' \in \mathcal{F}'$ such that $\|\nu(\chi) - \nu(\chi')\|^2$ is minimised
  - for some feature spaces the distance function is replaced by SAD or Hamming
- to remove an ambiguous matching we also find the second best match $\chi'_{\mathrm{sec}} \in \mathcal{F}'$ and calculate the differential ratio

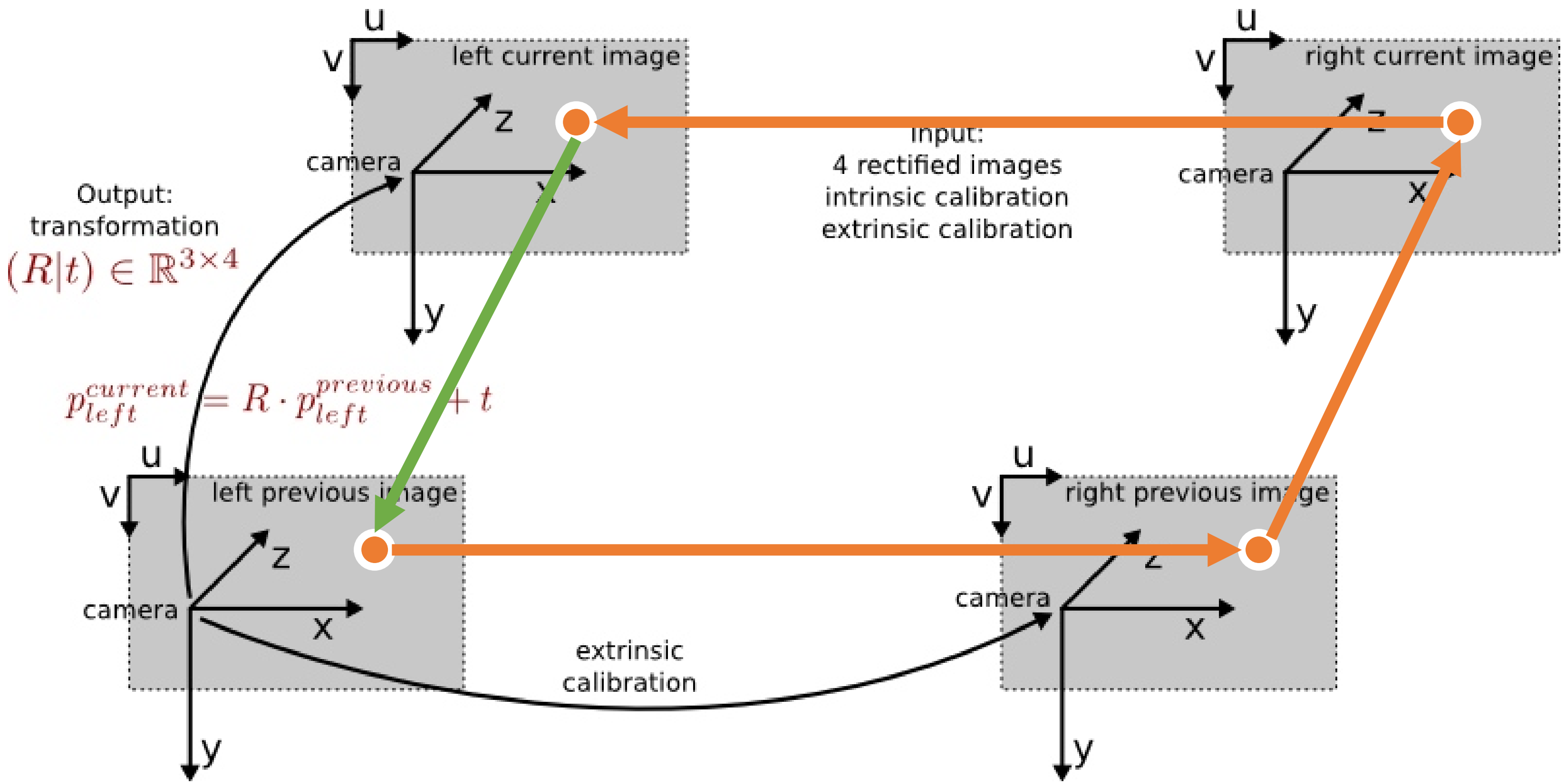
$$\varepsilon(\chi, \chi', \chi'_{\mathrm{sec}}; \nu) = \frac{\|\nu(\chi) - \nu(\chi')\|}{\|\nu(\chi) - \nu(\chi'_{\mathrm{sec}})\|}$$

- which will then be used to accept/reject matching $\chi \to \chi'$
  (note ε becomes very close to 1.0 in ambiguous case)

LIBVISO 1 & 2 (C++ Library for Visual Odometry)

Andreas Geiger et. al., MPI for Intelligent Systems in Tübingen, Germany, 2010-11.
648 citations

- Enhanced matching by cyclic check

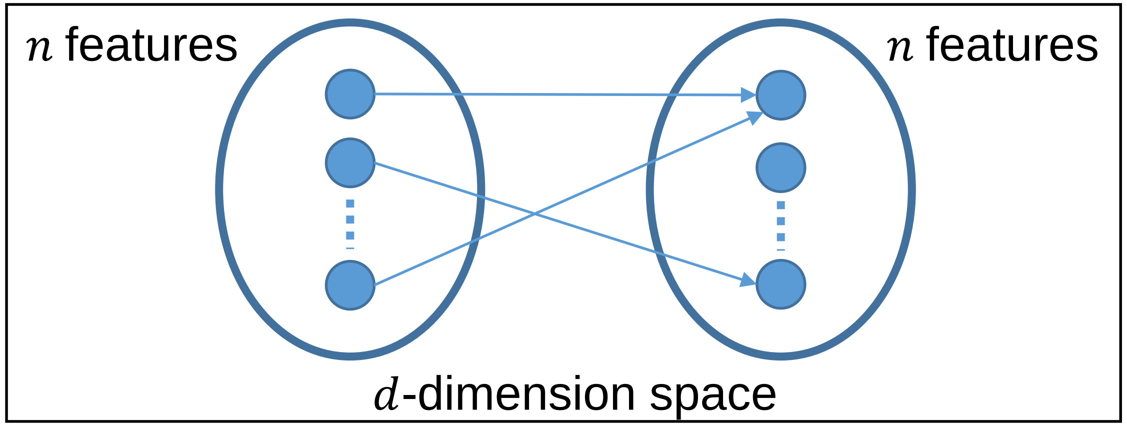


https://www.youtube.com/watch?v=DPLh6MoxPAk

Drawbacks

Why using image features can be BAD

- Image geometry and topology are not preserved in the feature space
- Direct matching on feature vectors may violate intrinsic constraints (e.g. Epipolar condition, ordering constraint)
- Need model-based outlier rejection schemes to ensure validity
  - RANSAC, M-SAC, LMedS, etc.
  - Non-deterministic
  - Iterative and time consuming
  - Convergence yet not guaranteed



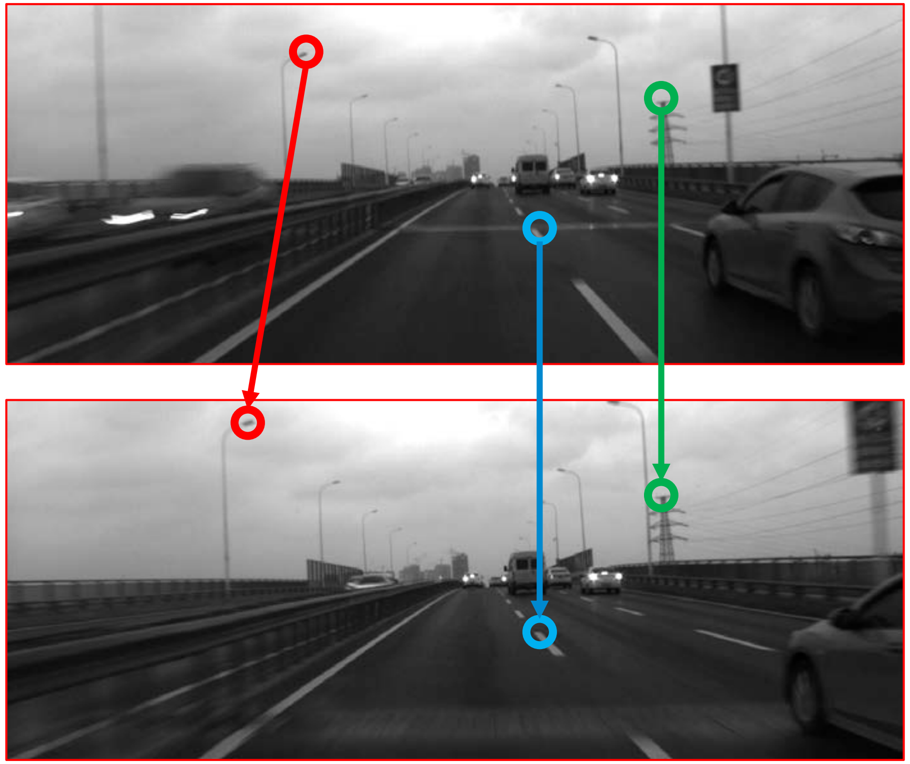
- Moreover..
  - Exact k-NN search in a high dimensional feature space is very expensive (the time complexity is $O(n^2d)$ given n d-vector features in each set)

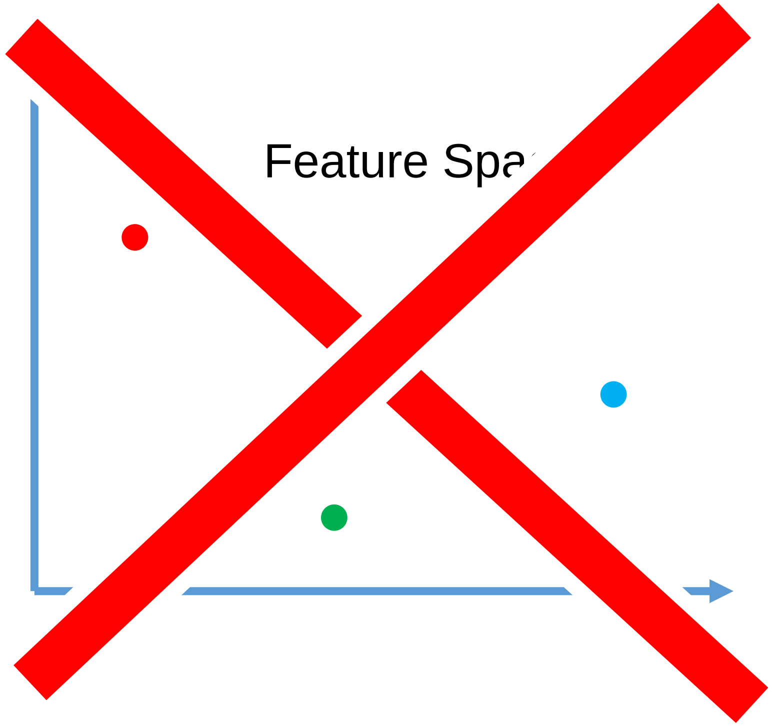
Direct method

Image Space

Frame t

Frame $t + 1$





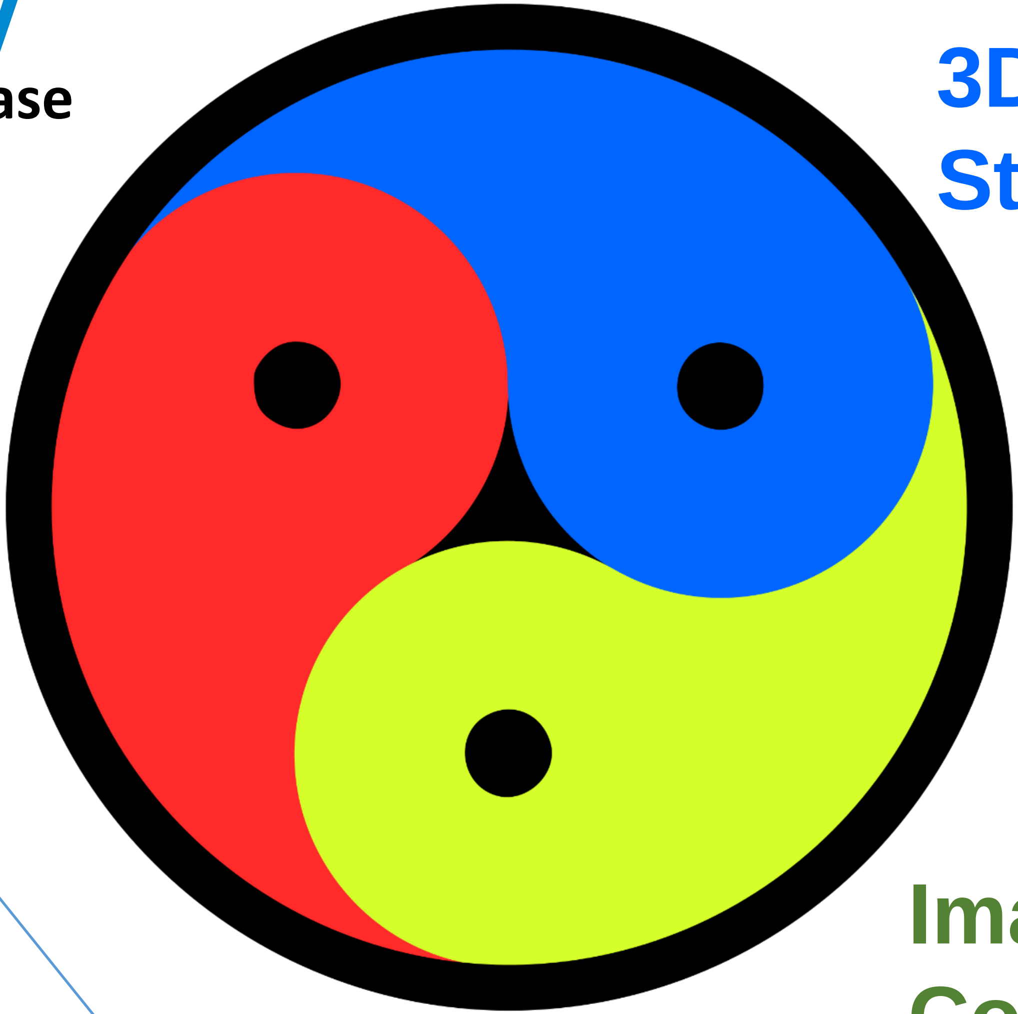
Note: This is NOT a general point tracking / optical flow problem. The scene structure and ego-motion need to be taken into account through the tracking process.

Featureless approach

- Make a full use of image intensities
- No feature space involved (thus no need to do feature transforms)
- Perform image warping & alignment to solve for camera motion
- No need to know **image correspondences**
  - Such correspondences are a by-product of the motion estimation process
- Need to know **scene structure** beforehand
  - Feature-based methods are able to estimate camera pose (up to a scale) directly from 2D-to-2D image correspondences, without any knowledge regarding scene structure

The Triality

in the featureless case



3D Scene Structure

given this

Camera Motion

Image Correspondences

find these

Demo

Final translational error:
5.89 cm (4.8%)
after 103 iterations

https://www.youtube.com/watch?v=6QNDsVfWqb4

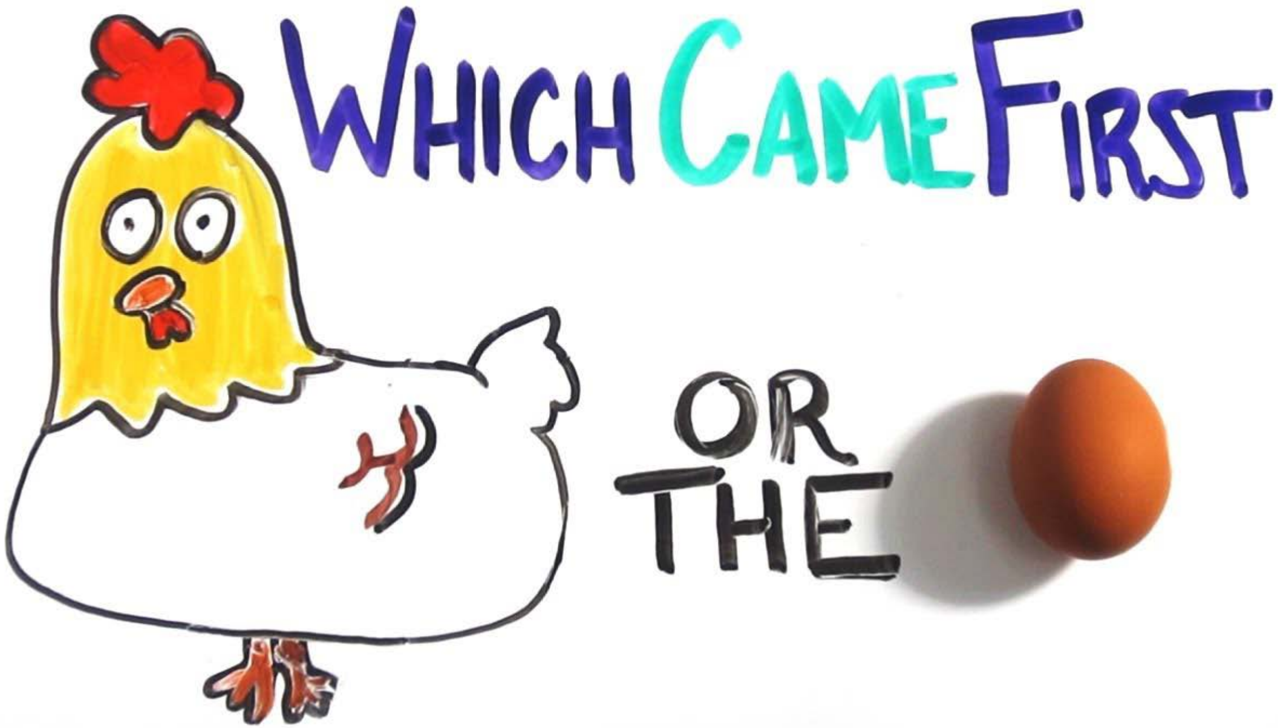
Summary by steps

- Indirect Methods
  1. Transform image pixels to a feature space
  2. Perform matching in feature space, with model-based outlier rejection
  3. Try an initial $(\mathbf{R}, \mathbf{t})$ and find each feature's projection in the next frame
  4. Compare the projected position with the matched feature
  5. Iteratively adjust $(\mathbf{R}, \mathbf{t})$ to lower such geometric distance (i.e. Φ_{geo})

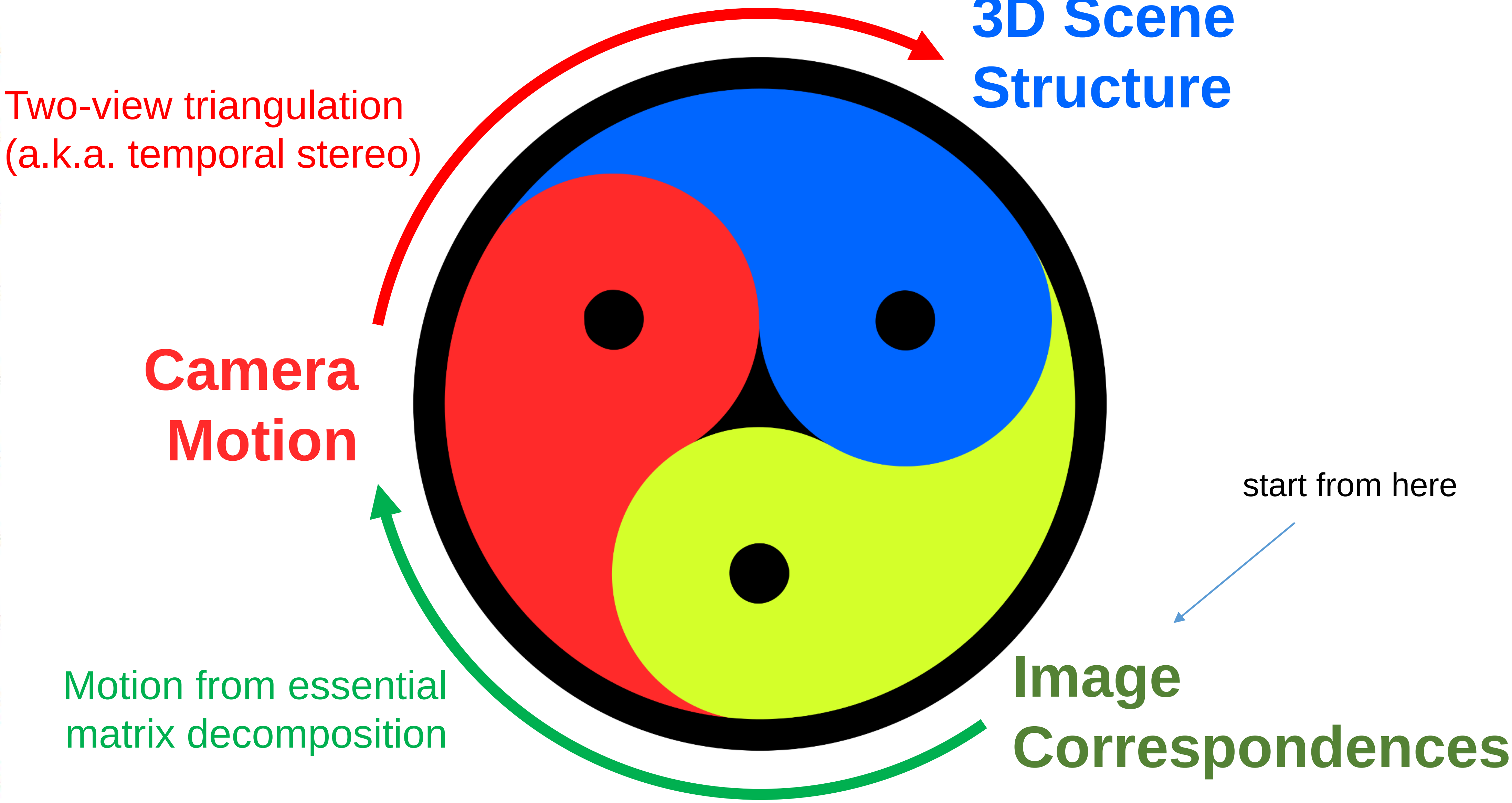
- Direct Methods
  1. Try an initial $(\mathbf{R}, \mathbf{t})$ to and find each pixel's projection in the next frame, given depth prior
  2. Compare the intensity of the projected pixel in the next frame with one in the current frame
  3. Iteratively adjust $(\mathbf{R}, \mathbf{t})$ to lower such photometric difference (i.e. Φ_{photo})

Monocular vision

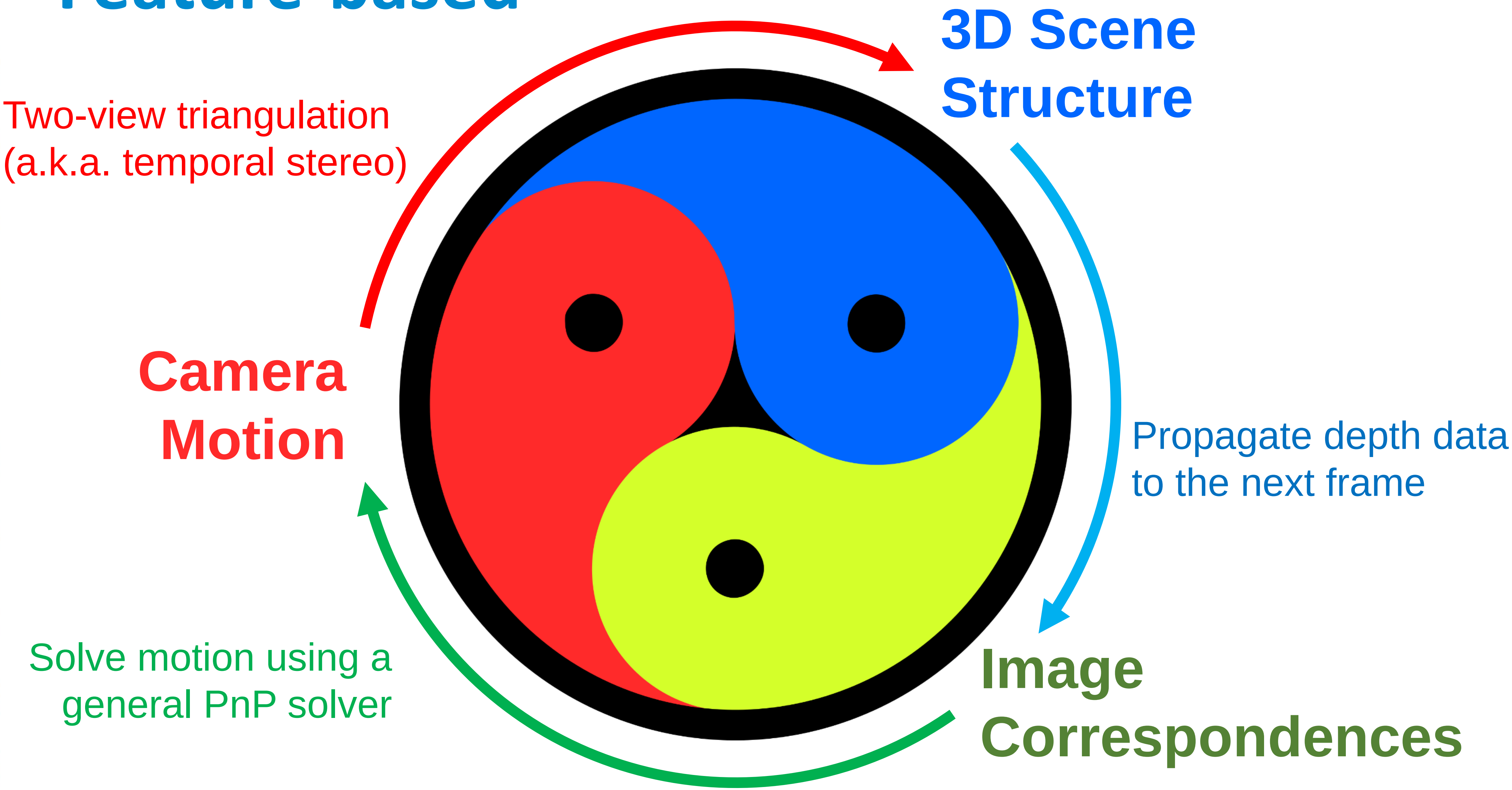
where the 3D reconstruction of scene structure is based on temporal stereo triangulation, which needs the ego-motion, which needs the 3D reconstruction of scene structure..



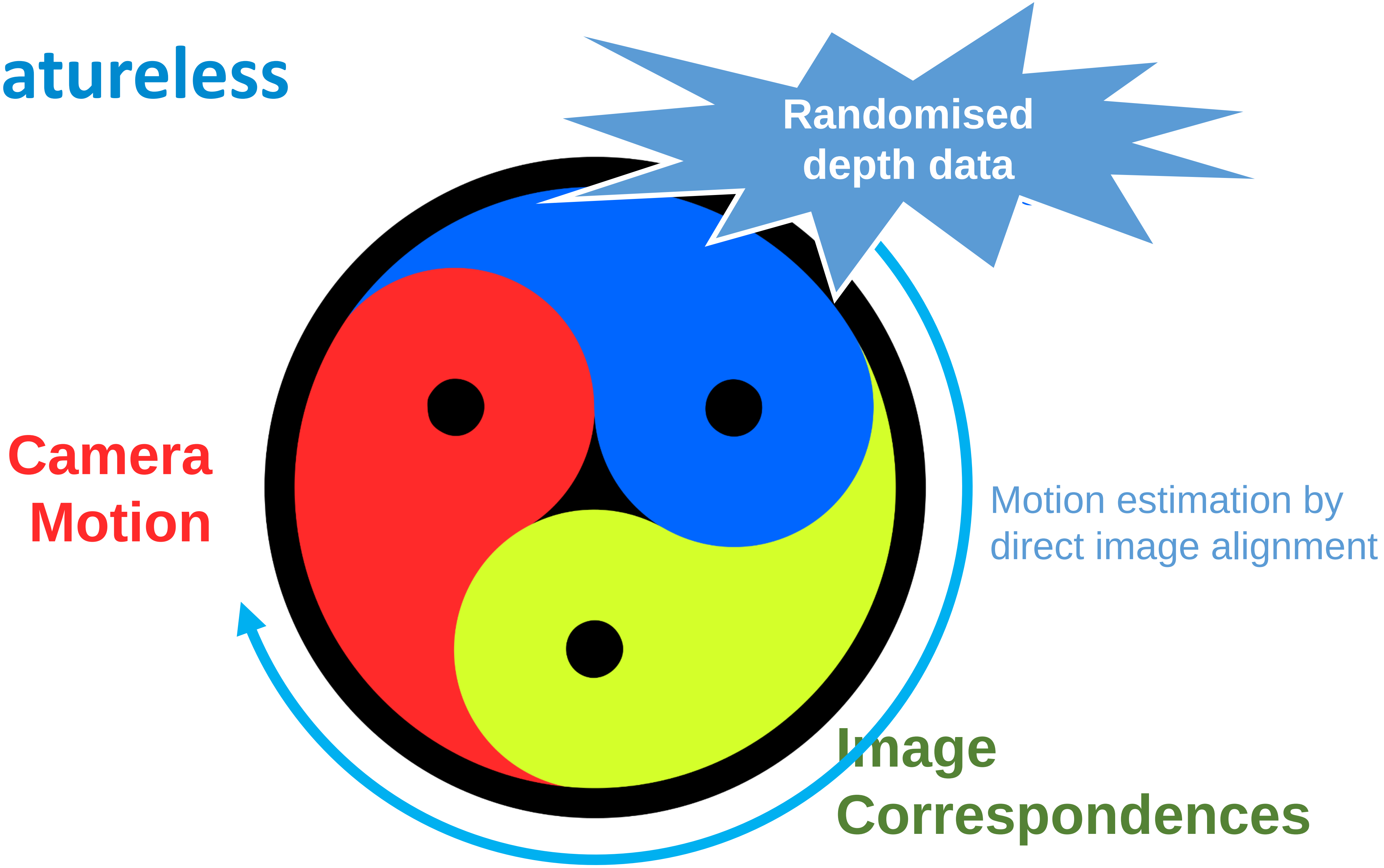
Feature-based



Feature-based

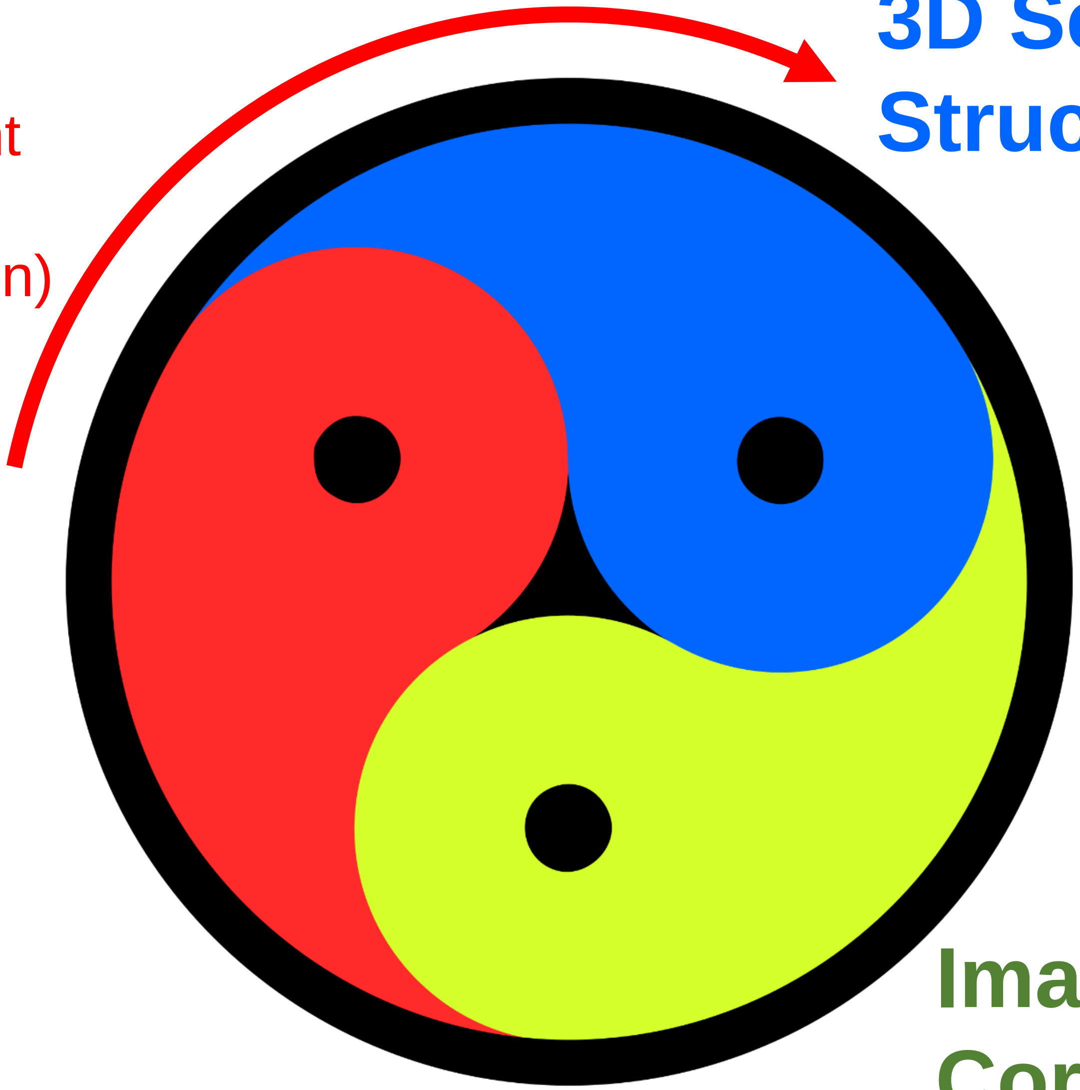


Featureless

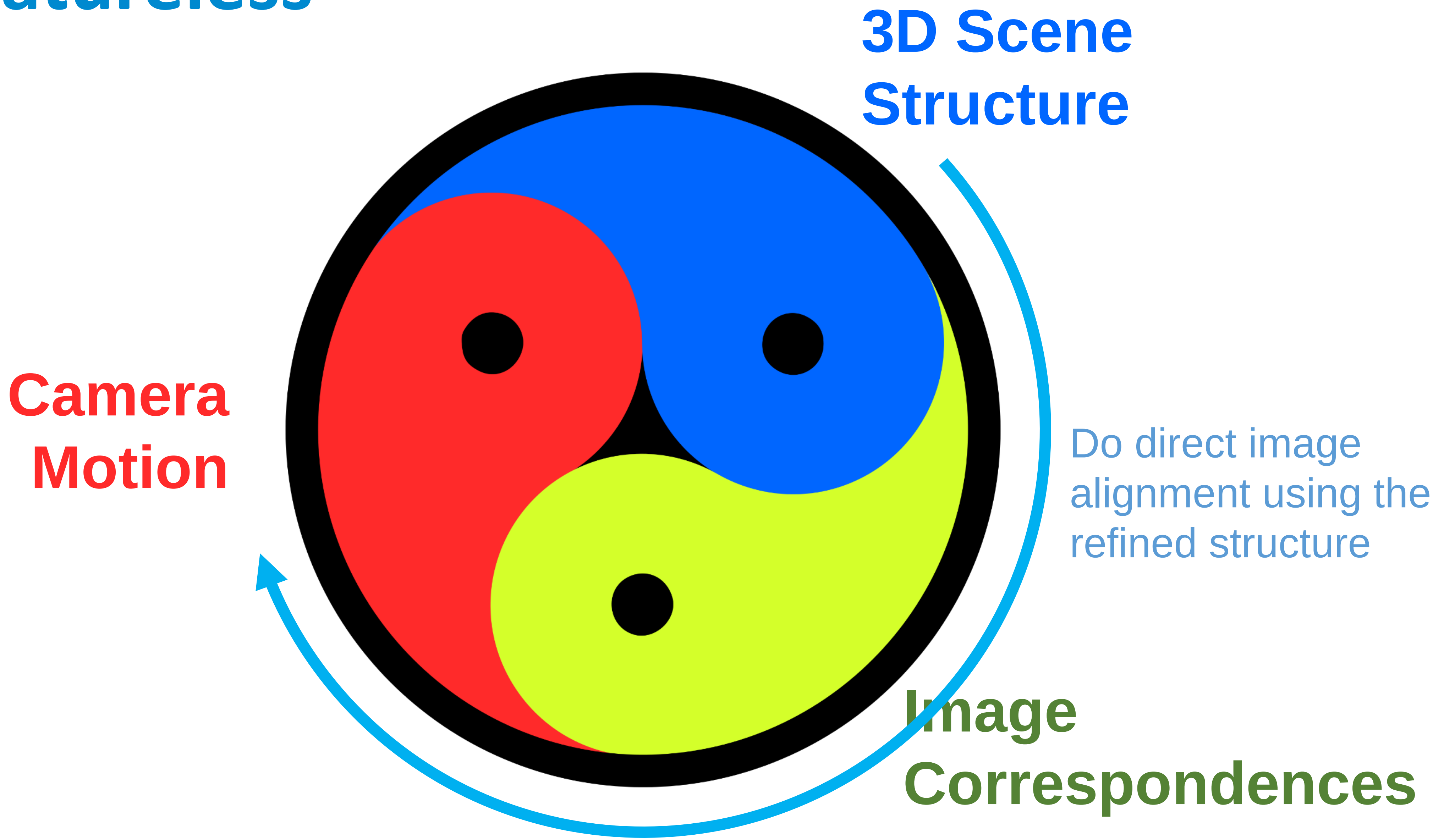


Featureless

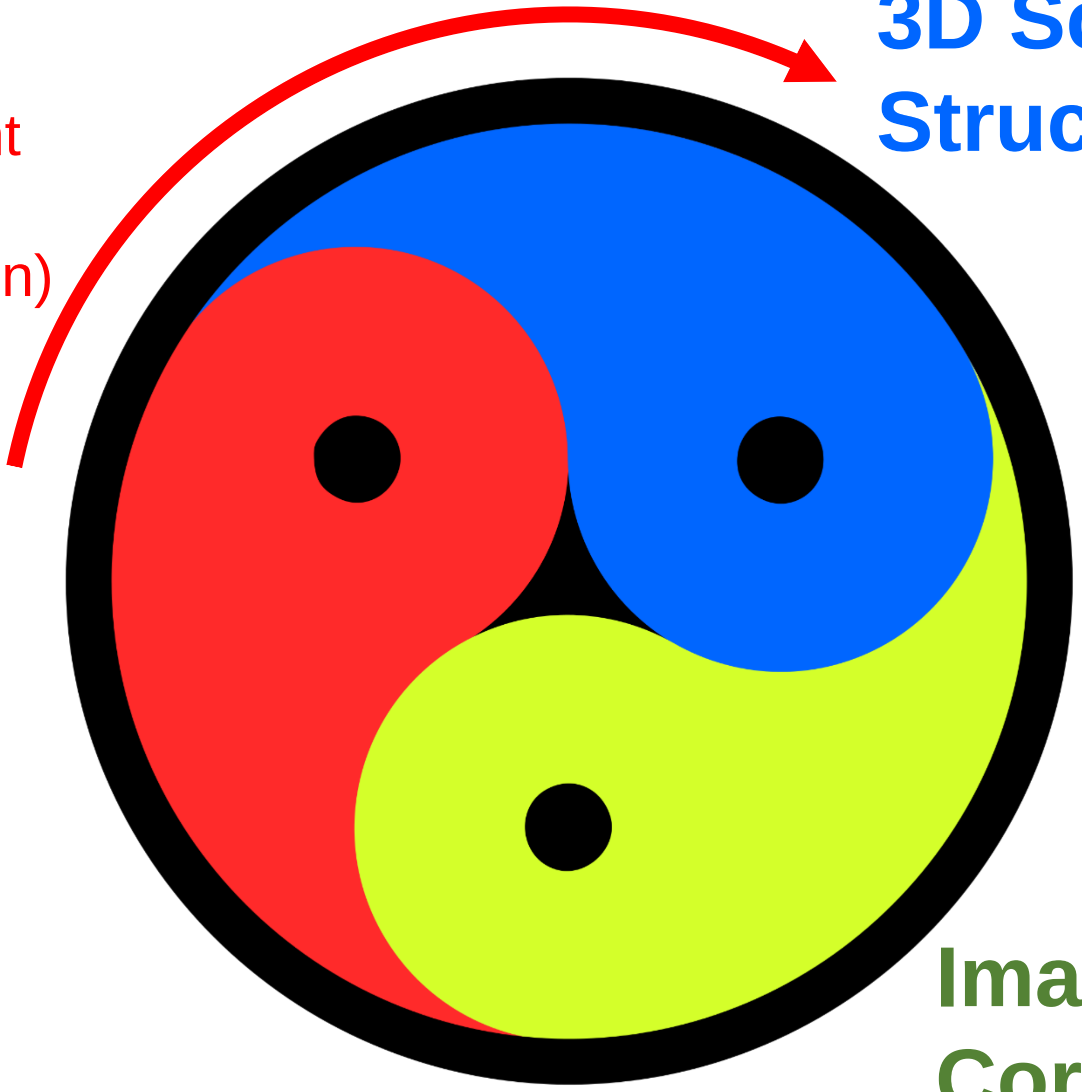
Structure refinement
by triangulation and
depth filtering (fusion)



Featureless



Featureless

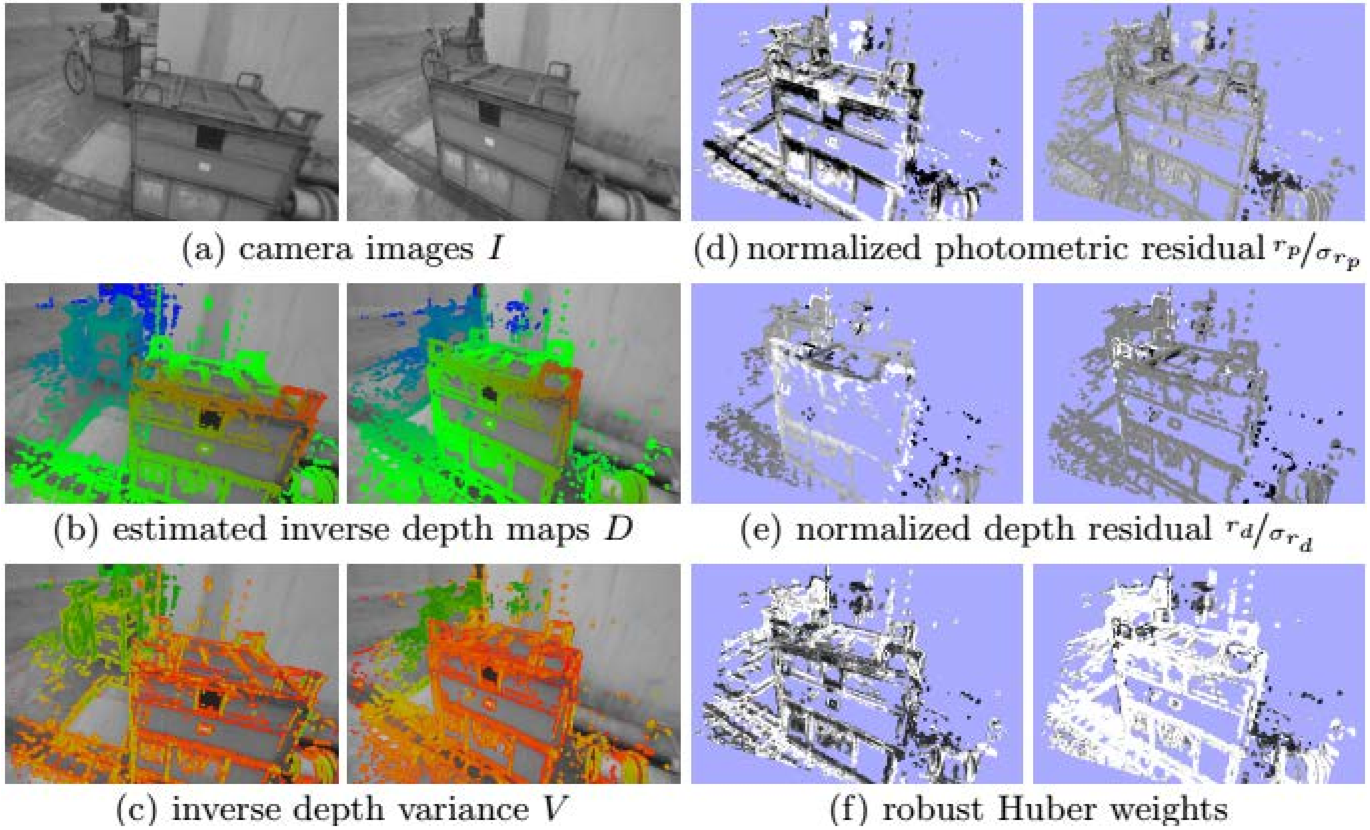


Large Scale Dense SLAM (LSD-SLAM)

Jakob Engel el. al., Technische Universität München (TUM), 2014-16

361 citations

- Semi-dense method
  - Use only edge pixels
- Take into account depth alignment errors



$$E(\boldsymbol{\xi}_{ji}) := \sum_{\mathbf{p}\in\Omega_{D_i}} \left\| \frac{r_p^2(\mathbf{p},\boldsymbol{\xi}_{ji})}{\sigma^2_{r_p(\mathbf{p},\boldsymbol{\xi}_{ji})}} + \frac{r_d^2(\mathbf{p},\boldsymbol{\xi}_{ji})}{\sigma^2_{r_d(\mathbf{p},\boldsymbol{\xi}_{ji})}} \right\|_\delta$$

$$r_d(\mathbf{p},\boldsymbol{\xi}_{ji}) := [\mathbf{p}']_3 - D_j([\mathbf{p}']_{1,2})$$

$$\mathbf{p}' := \omega_s(\mathbf{p}, D_i(\mathbf{p}), \boldsymbol{\xi}_{ji})$$

$$\sigma^2_{r_d(\mathbf{p},\boldsymbol{\xi}_{ji})} := V_j([\mathbf{p}']_{1,2}) \left(\frac{\partial r_d(\mathbf{p},\boldsymbol{\xi}_{ji})}{\partial D_j([\mathbf{p}']_{1,2})}\right)^2 + V_i(\mathbf{p}) \left(\frac{\partial r_d(\mathbf{p},\boldsymbol{\xi}_{ji})}{\partial D_i(\mathbf{p})}\right)^2$$

This plays a crucial role

https://www.youtube.com/watch?v=GnuQzP3gty4&t=9s

Drawbacks of the direct methods

- Dense matching is slow
  - as we compute and apply a homography (8-dof) for each feature's patch
  - lazy implementations often skip this
- Intensity alignment does not work for non-Lambertian and/or occluded surfaces
- Convergence not guaranteed
  - the image alignment process can diverge
  - especially in monocular case where scene structure initialisation is fully randomised
  - open issues remained

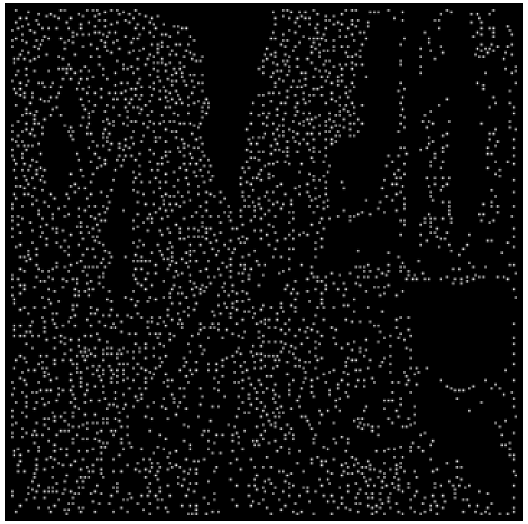
	Method	Setting	Code	Translation	Rotation	Runtime	Environment
[redacted]							
J. Zhang and S. Singh: Visual-lidar Odometry and Mapping: Low drift, Robust, and Fast. IEEE International Conference on Robotics and Automation(ICRA) 2015.							
[redacted]							
J. Zhang and S. Singh: LOAM: Lidar Odometry and Mapping in Real-time. Robotics: Science and Systems Conference (RSS) 2014.							
3	SOFT2			0.81 %	0.0022 [deg/m]	0.1 s	2 cores @ 2.5 Ghz (C/C++)
4	GDVO			0.86 %	0.0031 [deg/m]	0.09 s	1 core @ >3.5 Ghz (C/C++)
5	HypERROCC			0.88 %	0.0027 [deg/m]	0.25 s	2 cores @ 2.0 Ghz (C/C++)
6	SOFT			0.88 %	0.0022 [deg/m]	0.1 s	2 cores @ 2.5 Ghz (C/C++)
I. Cvišić and I. Petrović: Stereo odometry based on careful feature selection and tracking. European Conference on Mobile Robots (ECMR) 2015.							
7	RotRocc			0.88 %	0.0025 [deg/m]	0.3 s	2 cores @ 2.0 Ghz (C/C++)
M. Buczko and V. Willert: Flow-Decoupled Normalized Reprojection Error for Visual Odometry. 19th IEEE Intelligent Transportation Systems Conference (ITSC) 2016.							
8	EDVO			0.89 %	0.0030 [deg/m]	0.1 s	1 core @ 2.5 Ghz (C/C++)
9	svo2			0.94 %	0.0021 [deg/m]	0.2 s	1 core @ 2.5 Ghz (C/C++)
10	ROCC			0.98 %	0.0028 [deg/m]	0.3 s	2 cores @ 2.0 Ghz (C/C++)
M. Buczko and V. Willert: How to Distinguish Inliers from Outliers in Visual Odometry for High-speed Automotive Applications. IEEE Intelligent Vehicles Symposium (IV) 2016.							
11	cv4xv1-sc			1.09 %	0.0029 [deg/m]	0.145 s	GPU @ 3.5 Ghz (C/C++)
M. Persson, T. Piccini, R. Mester and M. Felsberg: Robust Stereo Visual Odometry from Monocular Techniques. IEEE Intelligent Vehicles Symposium 2015.							
[redacted]							
J. Zhang, M. Kaess and S. Singh: Real-time Depth Enhanced Monocular Odometry. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 2014.							
13	ORB-SLAM2		code	1.15 %	0.0027 [deg/m]	0.06 s	2 cores @ >3.5 Ghz (C/C++)
14	svo			1.16 %	0.0030 [deg/m]	0.1 s	2 core @ 2.5 Ghz (C/C++)
15	NOTF			1.17 %	0.0035 [deg/m]	0.45 s	1 core @ 3.0 Ghz (C/C++)
J. Deigmoeller and J. Eggert: Stereo Visual Odometry without Temporal Filtering. German Conference on Pattern Recognition (GCPR) 2016.							
16	S-PTAM		code	1.19 %	0.0025 [deg/m]	0.03 s	4 cores @ 3.0 Ghz (C/C++)
T. Pire, T. Fischer, J. Civera, P. Crist\'{o}foris and J. Jacobo-Berlles: Stereo parallel tracking and mapping for robot localization. IROS 2015.							
17	S-LSD-SLAM		code	1.20 %	0.0033 [deg/m]	0.07 s	1 core @ 3.5 Ghz (C/C++)
J. Engel, J. St\"uckler and D. Cremers: Large-Scale Direct SLAM with Stereo Cameras. Int.~Conf.~on Intelligent Robot Systems (IROS) 2015.							
18	VoBa			1.22 %	0.0029 [deg/m]	0.1 s	1 core @ 2.0 Ghz (C/C++)
J. Tardif, M. George, M. Laverne, A. Kelly and A. Stentz: A new approach to vision-aided inertial navigation. 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan 2010.							
19	SLUP			1.25 %	0.0041 [deg/m]	0.17 s	4 cores @ 3.3 Ghz (C/C++)
20	FRVO			1.26 %	0.0038 [deg/m]	0.03 s	1 core @ 3.5 Ghz (C/C++)

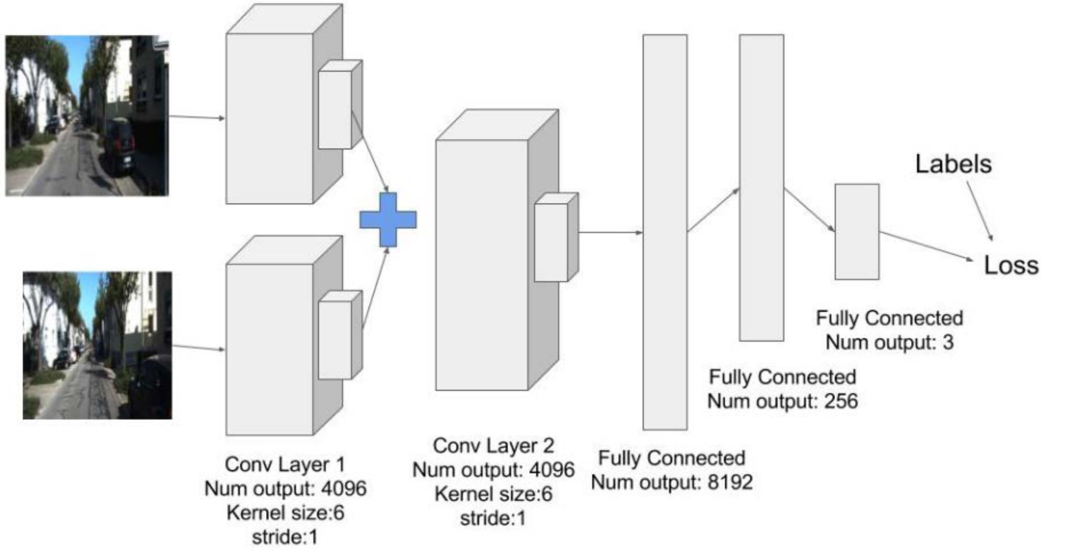
Visual-inertial approaches

- Incorporate inertial measurement into the ego-motion estimation stage
- Need covariance matrix modelling for data fusion
  - The matrix controls our “belief” in the quality of data from different sources

DL approaches..

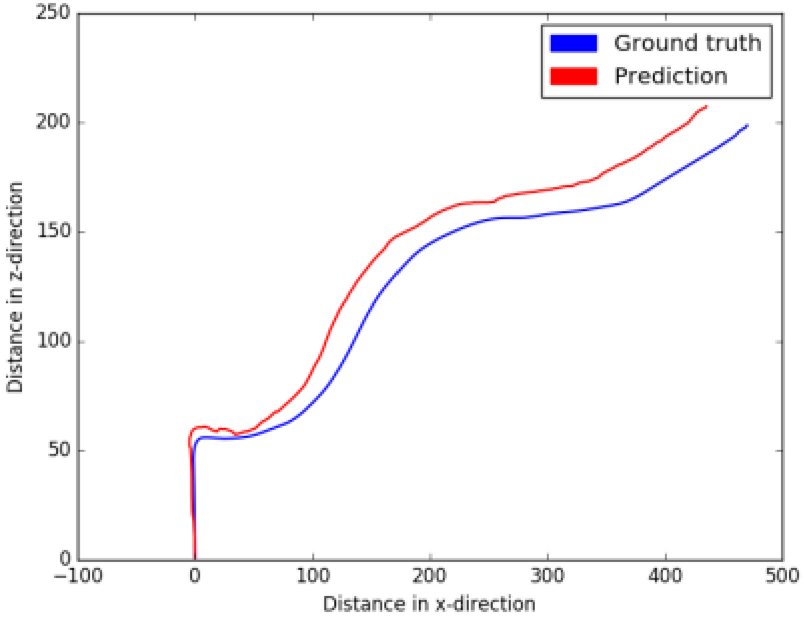
- DeepVO: A Deep Learning approach for Monocular Visual Odometry, Mohanty et. al, Nov. 2016.
  - Caffe used; the network design is heavily influenced by AlexNet
  - Adopted a simplified 3-dof planar motion model $(\Delta x, \Delta z, \Delta\theta)$
  - Input: RGB + a binary FAST image

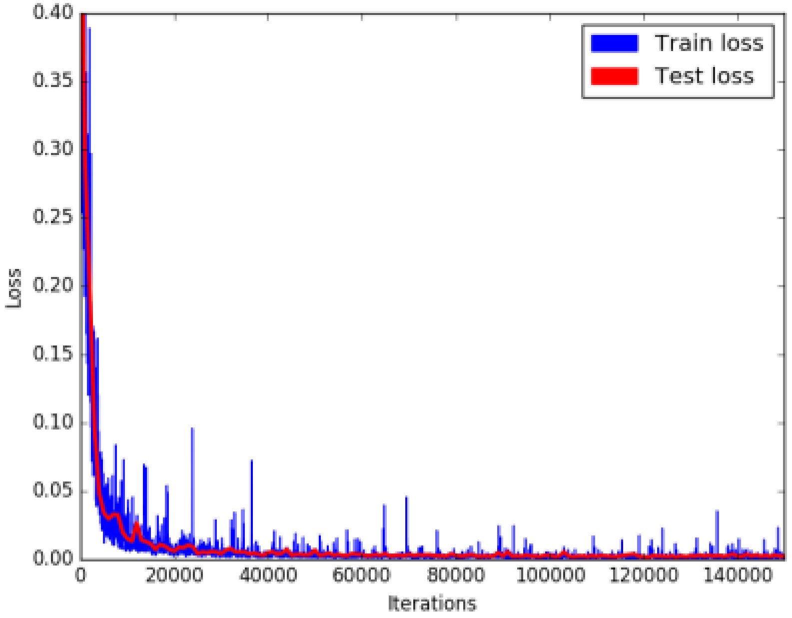




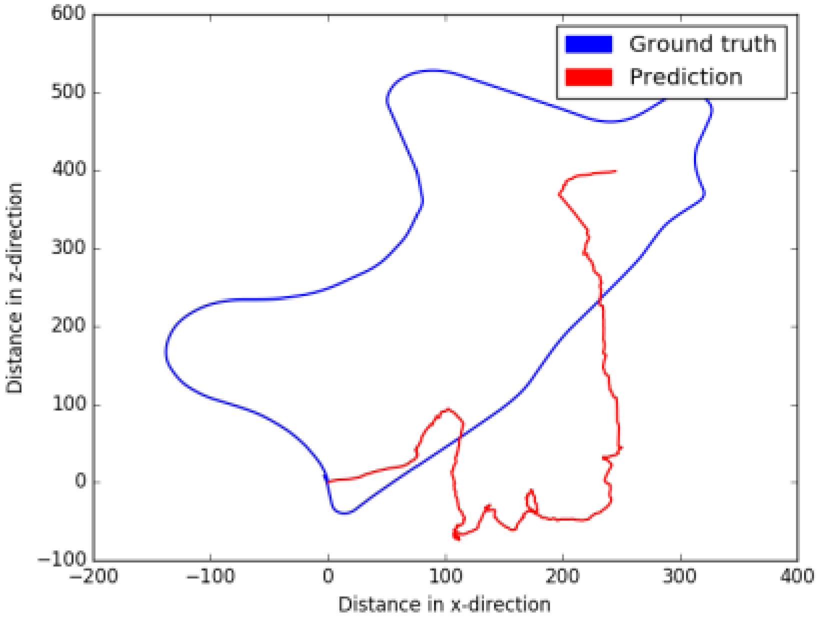
Results

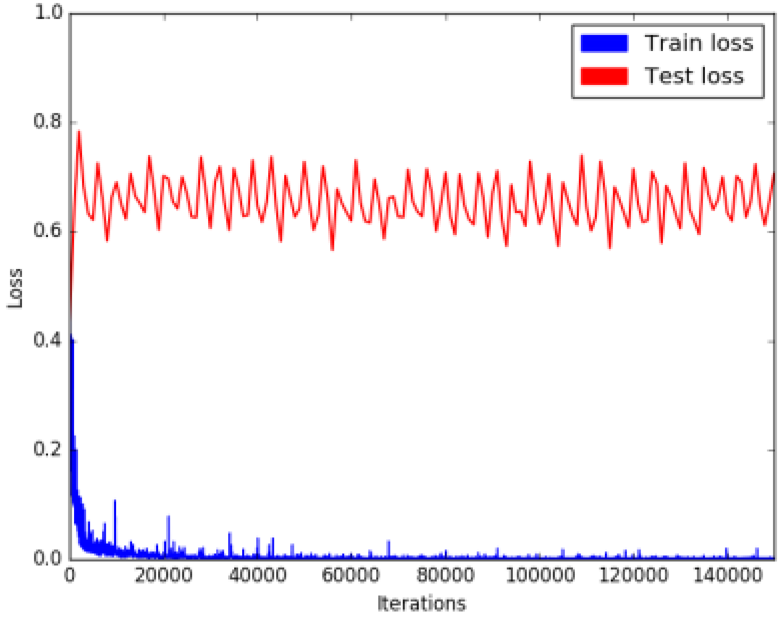
50-50 Training and testing





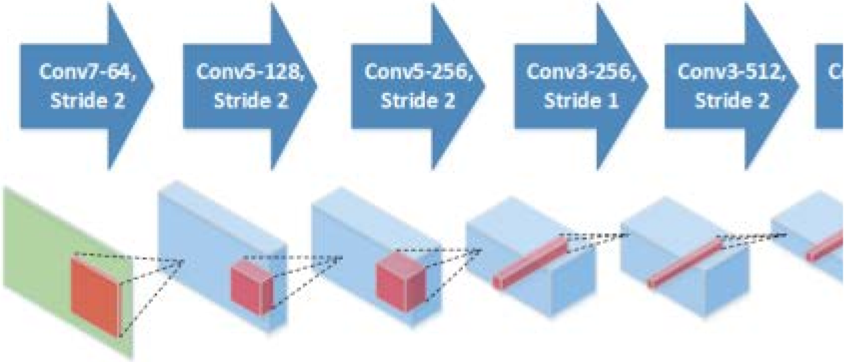
Completely unknown scene

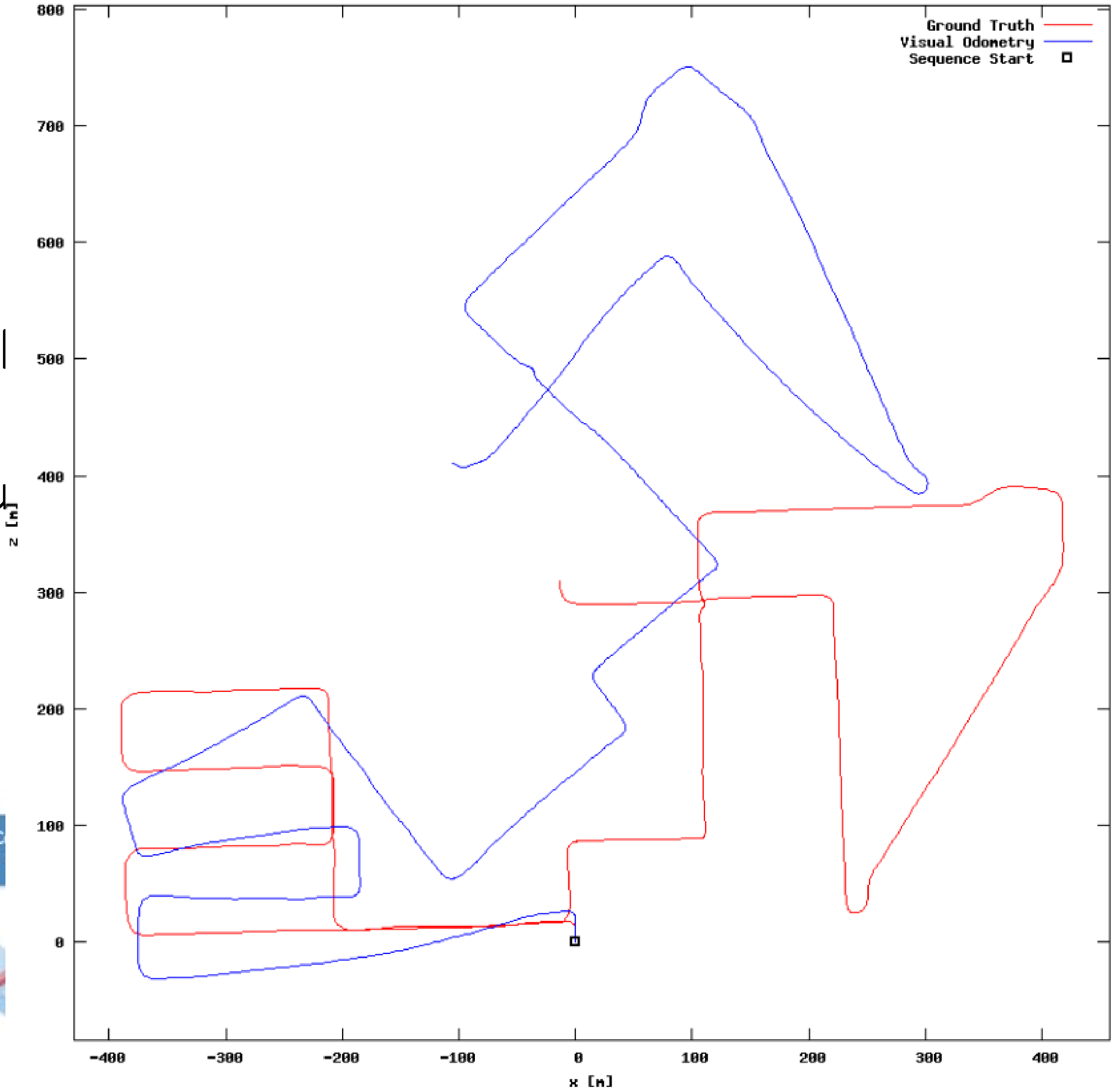




DL approaches

- Optical Flow and Deep Lea
  to Visual Odometry, Peter
    - Based on Caffee
    - Use optical flow as the inpu





Summary

- The paradigm is shifting
  - from sparse pixels to dense
  - from stereo vision to monocular
  - from ground into the sky
  - from high-end devices to pervasive
  - from individual approach to hybrid
- Few real-time implementations available
  - ICCV (December, 2015) features a Future of Real-Time SLAM Workshop
- Few attempts on DL approaches
  - far from acceptable accuracy (<2%)

Videos and content

- Courtesy of Johnny Chien
- https://www.youtube.com/watch?v=6QNDsVfWqb4
- https://www.youtube.com/watch?v=q3fIeO34cKE