Synchronization and Cycle-Consistency: Application to Motion Segmentation and Localization

Federica Arrigoni

DISI – University of Trento (Italy)

federica.arrigoni@unitn.it

Tutorial on "Cycle Consistency and Synchronization in Computer Vision"

In Conjunction with CVPR 2020

Outline

1. Synchronization & Motion Segmentation
   - Introduction: synchronization, spectral methods and applications
   - Motion segmentation: spectral method
   - Motion segmentation: local method
   - Concluding remarks

2. Cycle consistency & Localization
   - Introduction: translation synchronization and localization problems
   - Bearing-based localization and parallel rigidity
   - Application to structure from motion
   - Concluding remarks

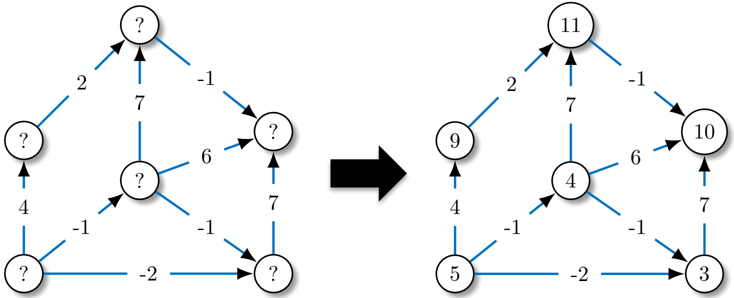
Outline

1. Synchronization & Motion Segmentation
   - Introduction: synchronization, spectral methods and applications
   - Motion segmentation: spectral method
   - Motion segmentation: local method
   - Concluding remarks

2. Cycle consistency & Localization
   - Introduction: translation synchronization and localization problems
   - Bearing-based localization and parallel rigidity
   - Application to structure from motion
   - Concluding remarks

Time Synchronization

Consider a network of nodes where each node is characterized by an unknown state and pairs of nodes can measure the **difference** between their states.

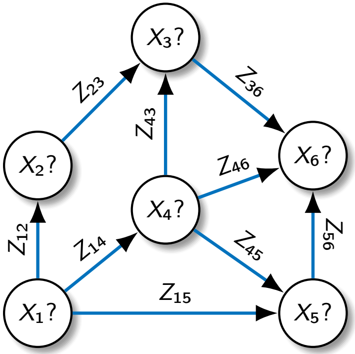


The goal is to infer the unknown states from the pairwise measures.

A. Giridhar and P. Kumar Distributed clock synchronization over wireless networks: Algorithms and analysis
IEEE Conference on Decision and Control (2006)

Synchronization

The goal of synchronization is to recover elements of a **group**, given a certain number of their mutual differences (or **ratios**).



Consistency Constraint

$$Z_{ij} = X_i \cdot X_j^{-1}$$

Ambiguity

$$Z_{ij} = (X_i \cdot S) \cdot (X_j \cdot S)^{-1}$$

for any (fixed) S

The problem is well-posed only if the underlying graph is **connected**.

Spectral Solution

If the group admits a matrix representation, the unknown elements can be recovered via **spectral decomposition**.

$$Z_{ij} = X_i X_j^{-1} \quad \Longrightarrow \quad \boxed{ZX = nX}$$

$$Z = \begin{bmatrix} I & Z_{12} & \dots & Z_{1n} \\ Z_{21} & I & \dots & Z_{2n} \\ \dots & & & \dots \\ Z_{n1} & Z_{n2} & \dots & I \end{bmatrix}, \quad X = \begin{bmatrix} X_1 \\ X_2 \\ \dots \\ X_n \end{bmatrix}$$

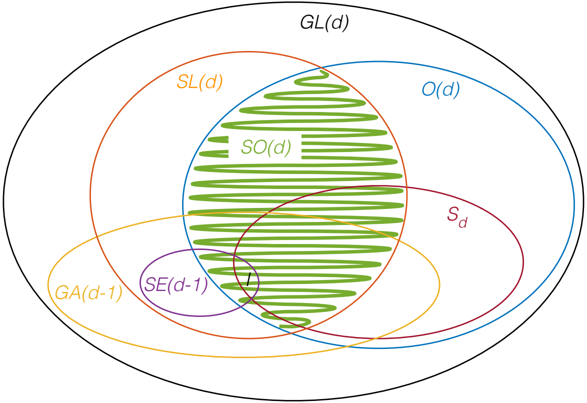
In order to fix the group **ambiguity**, the first element can be set to the identity.
At the end the solution is **projected** onto the group.

Extensions

- Missing data $\longrightarrow$ adjacency/degree matrix
- Outliers $\longrightarrow$ iteratively reweighted least squares (IRLS)

Examples

The spectral solution has been applied to several **matrix groups** in Computer Vision literature.



F. Arrigoni and A. Fusiello Synchronization problems in Computer Vision with closed-form solutions IJCV (2020)

Example: rotations

$$SO(d) = \{M \in \mathbb{R}^{d\times d} s.t.\ M^{\mathsf{T}}M = MM^{\mathsf{T}} = I_d,\ \det(M) = 1\}$$

Application: structure from motion

Projection: singular value decomposition

J. Keller Closest unitary, orthogonal and Hermitian operators to a given operator. Mathematics Magazine (1975)

A. Singer Angular synchronization by eigenvectors and semidefinite programming Applied and Computational Harmonic Analysis (2011)

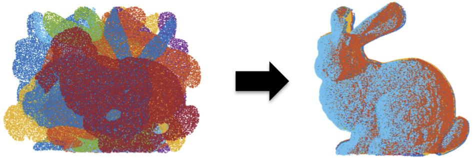
A. Singer and Y. Shkolnisky Three-dimensional structure determination from common lines in cryo-EM by eigenvectors and semidefinite programming SIAM Journal on Imaging Sciences (2011)

M. Arie-Nachimson, S.Z. Kovalsky, I. Kemelmacher-Shlizerman, A. Singer and R. Basri Global motion estimation from point matches Joint 3DIM/3DPVT Conference (2012)

Example: rigid motions

$$SE(d) = \left\{ \begin{bmatrix} M & \mathbf{t} \\ \mathbf{0}^\top & 1 \end{bmatrix},\ s.t.\ M \in SO(d),\ \mathbf{t} \in \mathbb{R}^d \right\}$$

Application: point-cloud registration



Projection: singular value decomposition

C. Belta and V. Kumar Euclidean metrics for motion generation on SE(3) Journal of Mechanical Engineering Science (2002)

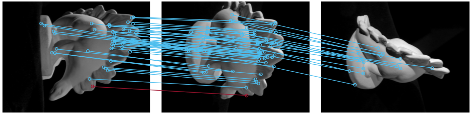
F. Bernard, J. Thunberg, P. Gemmar, F. Hertel, A. Husch, and J. Goncalves A solution for multi-alignment by transformation synchronisation. CVPR (2015)

F. Arrigoni, B. Rossi and A. Fusiello Spectral synchronization of multiple views in SE(3) SIAM Journal on Imaging Sciences (2016)

Example: permutations

$$\mathcal{S}_d = \{M \in \{0,1\}^{d\times d}\ s.t.\ M\mathbf{1} = \mathbf{1},\ \mathbf{1}M = \mathbf{1}\}$$

Application: multi-view matching



Projection: linear assignment problem

H. W. Kuhn The Hungarian method for the assignment problem Naval Research Logistics Quarterly (1955)

D. Pachauri, R. Kondor, and V. Singh Solving the multi-way matching problem by permutation synchronization. NIPS (2013)

Y. Shen, Q. Huang, N. Srebro and S. Sanghavi Normalized spectral map synchronization. NIPS (2016)

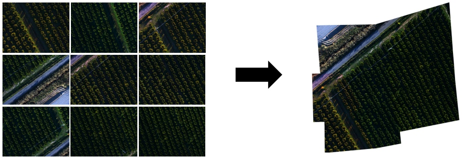
E. Maset, F. Arrigoni and A. Fusiello Practical and efficient multi-view matching ICCV (2017)

F. Bernard, J. Thunberg, P. Swoboda, C. Theobalt HiPPI: Higher-Order Projected Power Iterations for Scalable Multi-Matching ICCV (2019)

Example: homographies

$$SL(d) = \{M \in \mathbb{R}^{d\times d} \ s.t. \ \det(M) = 1\}$$

Application: image mosaicking



Projection: divide by $\sqrt[d]{det}$ (d odd)

E. Malis and R. Cipolla Camera self-calibration from unknown planar structures enforcing the multiview constraints between collineations PAMI (2002)

P. Schroeder, A. Bartoli, P. Georgel and N. Navab Closed-form solutions to multiple-view homography estimation. WACV (2011)

E. Santellani, E. Maset, and A. Fusiello Seamless image mosaicking via synchronization ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences (2018)

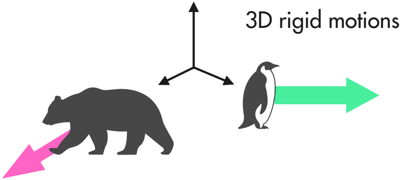
Outline

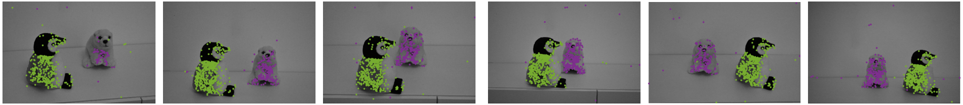
1. Synchronization & Motion Segmentation
   - Introduction: synchronization, spectral methods and applications
   - Motion segmentation: spectral method
   - Motion segmentation: local method
   - Concluding remarks

2. Cycle consistency & Localization
   - Introduction: translation synchronization and localization problems
   - Bearing-based localization and parallel rigidity
   - Application to structure from motion
   - Concluding remarks

Application to Motion Segmentation

The goal is to classify points in multiple images based on the **moving object** they belong to.

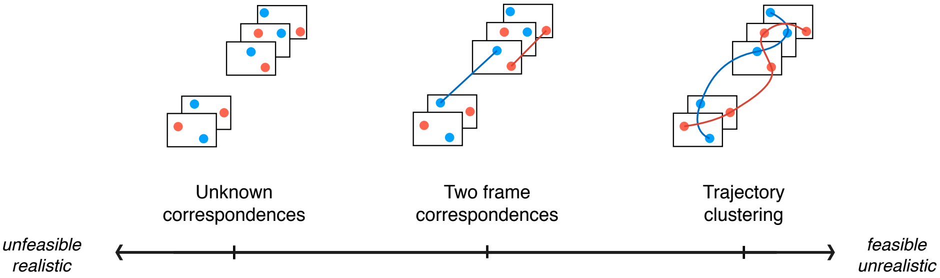




M. R. U. Saputra, A. Markham, and N. Trigoni Visual SLAM and structure from motion in dynamic environments: a survey ACM Computing Surveys (2018)

Application to Motion Segmentation

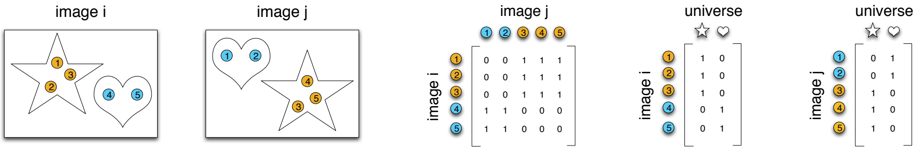
Input: two-frame correspondences (i.e. **pairwise matches**) $\longrightarrow$ good trade-off between making realistic assumptions and addressing a feasible task



Assumption: the **number of motions** is known.

Problem Formulation

Motion segmentation can be seen as a synchronization of **binary** matrices.



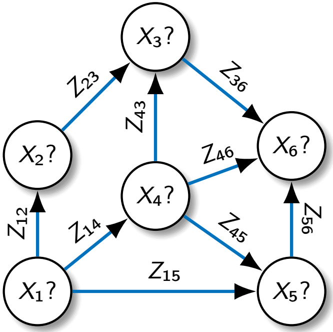
Partial segmentation: **local/relative** representation

- $[Z_{ij}]_{h,k} = 1$ if point h in image i and point k in image j belong to the same motion;
- $[Z_{ij}]_{h,k} = 0$ otherwise.

Total segmentation: **global/absolute** representation

- $[X_i]_{h,k} = 1$ if point h in image i belongs to motion k;
- $[X_i]_{h,k} = 0$ otherwise.

Problem Formulation



Consistency constraint

$$Z_{ij} = X_i X_j^{\mathsf{T}}$$

1. Each partial segmentation is computed by fitting multiple **fundamental matrices** to corresponding points.

   L. Magri and A. Fusiello Robust multiple model fitting with preference analysis and low-rank approximation. BMVC (2015)

2. Total segmentations are derived via synchronization (**spectral solution**).

Spectral Solution

Consistency constraint $\boxed{Z = XX^\mathsf{T}} \implies \operatorname{rank}(Z) = d$

$$\textbf{input} \longleftarrow \quad Z = \begin{bmatrix} Z_{11} & Z_{12} & \dots & Z_{1n} \\ Z_{21} & Z_{22} & \dots & Z_{2n} \\ \dots & & & \dots \\ Z_{n1} & Z_{n2} & \dots & Z_{nn} \end{bmatrix} \qquad X = \begin{bmatrix} X_1 \\ X_2 \\ \dots \\ X_n \end{bmatrix} \quad \longrightarrow \textbf{output}$$

Remark. $X_i^\mathsf{T} X_i$ is a $d \times d$ diagonal matrix s.t. the (k,k)-entry counts the number of points in image i that belong to motion k. Thus $X^\mathsf{T} X = \sum_{i=1}^n (X_i^\mathsf{T} X_i)$ is a diagonal matrix s.t. (k,k)-entry counts the number of points over all the images that belong to motion k.

$$ZX = XX^\mathsf{T}X = X \underbrace{\sum_{i=1}^{n} (X_i^\mathsf{T} X_i)}_{\Lambda} = X \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_d \end{bmatrix}$$

Spectral Solution

Proposition

*The columns of X are d (orthogonal) eigenvectors of Z, where d is the number of motions. If each motion contains a **different** number of points, then the eigenvalues are distinct.*

Algorithm - Synch

1. Compute the top d eigenvectors of Z, collected in a matrix U.
2. Transform U into a binary matrix:
   - $[X]_{h,k} = 1$ if the following conditions are satisfied:
     a) $[U]_{h,k}$ is the maximum value over row h;
     b) $[U]_{h,k} \neq 0$;
     c) $[U]_{h,k}/[U]_{h,l} > \theta$ where $[U]_{h,l}$ is the 2nd-maximum value over row h;
   - $[X]_{h,k} = 0$ otherwise.

F. Arrigoni and T. Pajdla Motion segmentation via synchronization ICCV Workshops (2019)

Optimization problem

Cost function: the number of points equally labelled by the partial segmentations Z_{ij} and $X_i X_j^\mathsf{T}$ are counted for each image pair.

$$\max_{X_1,\ldots,X_n} \sum_{i,j=1}^{n} \operatorname{trace}(Z_{ij}^\mathsf{T} X_i X_j^\mathsf{T}) \quad \Longleftrightarrow \quad \max_{X} \operatorname{trace}(X^\mathsf{T} Z X)$$

$$\text{s.t. } X_i \in \{0,1\}^{p_i \times d},\ X_i \mathbf{1} = \mathbf{1} \quad \forall\, i = 1,\ldots,n \qquad\qquad \text{s.t. } X \in \{0,1\}^{p \times d},\ X\mathbf{1} = \mathbf{1}$$

Spectral relaxation: the optimization variable is treated as a real matrix instead of a binary matrix, and its columns are enforced to be orthonormal.

$$\max_{U} \operatorname{trace}(U^\mathsf{T} Z U) \quad \text{s.t. } U \in \mathbb{R}^{p \times d},\ U^\mathsf{T} U = I_d$$

This is a **Rayleigh problem**: the solution is given by the d leading eigenvectors of Z.

Outline

1. Synchronization & Motion Segmentation
   - Introduction: synchronization, spectral methods and applications
   - Motion segmentation: spectral method
   - Motion segmentation: local method
   - Concluding remarks

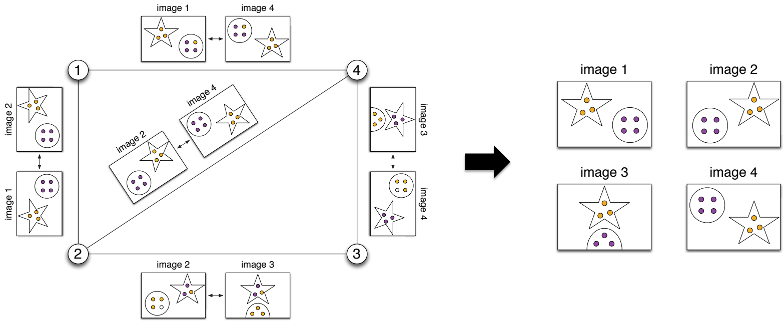
2. Cycle consistency & Localization
   - Introduction: translation synchronization and localization problems
   - Bearing-based localization and parallel rigidity
   - Application to structure from motion
   - Concluding remarks

Local solution

1. Segmentation of corresponding points is performed on each image pair in isolation (e.g. by fitting multiple **fundamental matrices**).

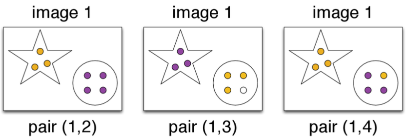
   L. Magri and A. Fusiello Robust multiple model fitting with preference analysis and low-rank approximation. BMVC (2015)

2. The partial results derived in Step 1 are combined in order to get a **multi-frame** segmentation.

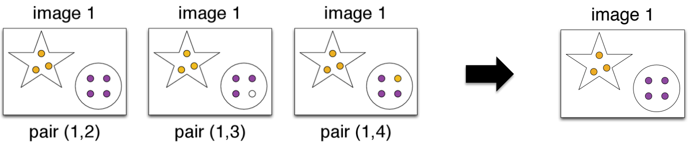


Local solution

- Idea: all the two-frame segmentations involving a fixed image provide (up to a permutation) an estimate for the segmentation of that image.

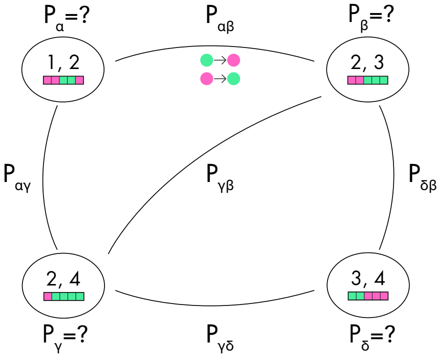


- The permutation ambiguity can be solved via **permutation synchronization** (next slide).
- For each point in each image, several putative labels are available: the most frequent label (**mode**) is chosen.



Local solution

Graph representation: each vertex corresponds to one pair of images; an edge is present between two vertices if and only if the associated pairs have one image in common.



- Compute a permutation for each edge: **linear assignment problem**

  H. W. Kuhn The Hungarian method for the assignment problem Naval Research Logistics Quarterly 2 (1955)

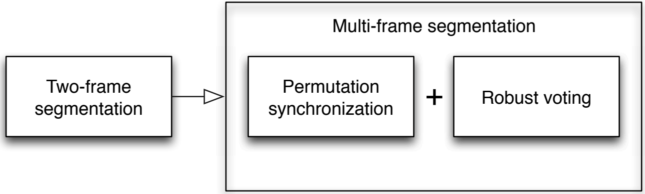
- Compute permutations for all the nodes: **permutation synchronization**

  D. Pachauri, R. Kondor, and V. Singh Solving the multi-way matching problem by permutation synchronization NIPS (2013)

Remark: the size of permutation matrices is equal to the **number of motions** (known).

Local solution

Algorithm - Mode



F. Arrigoni and T. Pajdla Robust motion segmentation from pairwise matches ICCV (2019)

Remark: this approach is **local**, as each image is segmented based on its neighbors only. Local solutions appeared also in the case of rotations and rigid-motions:

R. Hartley, K. Aftab and J. Trumpf L1 rotation averaging using the Weiszfeld algorithm. CVPR (2011)

A. Torsello, E. Rodolá and A. Albarelli Multiview registration via graph diffusion of dual quaternions CVPR (2011)

Experiments - Trajectory Clustering

Hopkins155 – Misclassification Error [%]

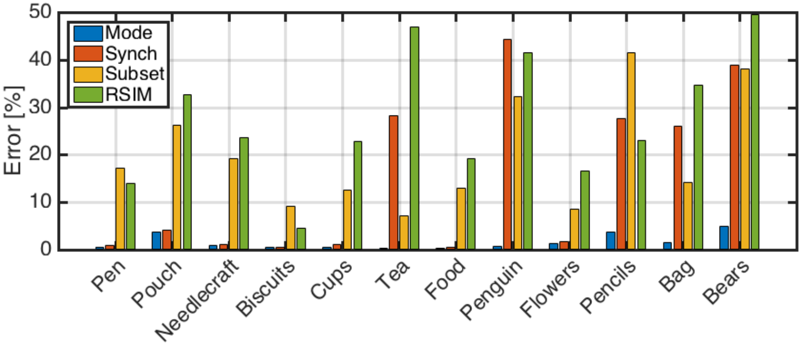
	LSA	GPCA	ALC	SSC	TPV	LRR	T-Linkage	S^3C	RSIM	MSSC	KerAdd	Coreg	Subset	Synch	Mode
2 Motions	4.23	4.59	2.40	1.52	1.57	1.33	0.86	1.94	0.78	0.54	0.27	0.37	**0.23**	2.70	1.00
3 Motions	7.02	28.66	6.69	4.40	4.98	4.98	5.78	4.92	1.77	1.84	0.66	0.75	**0.58**	6.99	2.67
All	4.86	10.02	3.56	2.18	2.34	1.59	1.97	2.61	1.01	0.83	0.36	0.46	**0.31**	3.67	1.37

Hopkins12 – Misclassification Error [%]

	PF	PF+ALC	RPCA+ALC	ℓ_1+ALC	SSC-R	SSC-O	RSIM	KerAdd	Coreg	Subset	Synch	Mode
Mean	14.94	10.81	13.78	1.28	3.82	8.78	0.61	0.11	**0.06**	**0.06**	5.46	4.33
Median	9.31	7.85	8.27	1.07	0.31	4.80	0.61	**0.00**	**0.00**	**0.00**	0.57	0.38

http://www.vision.jhu.edu/data/hopkins155/

Experiments - Pairwise Matches

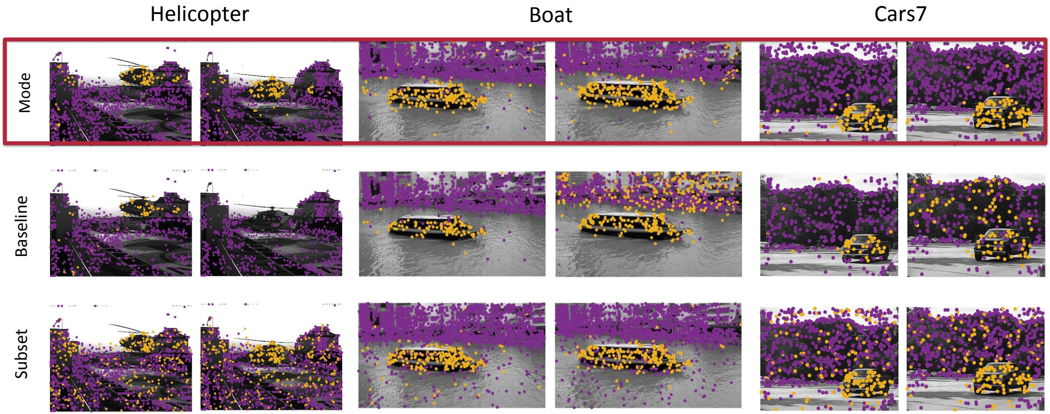


https://github.com/federica-arrigoni/ICCV_19

X. Xu, L.F. Cheong and Z. Li Motion segmentation by exploiting complementary geometric models CVPR (2018)

P. Ji, M. Salzmann, and H. Li Shape interaction matrix revisited and robustified: efficient subspace clustering with corrupted and incomplete data ICCV (2015)

Experiments - Pairwise Matches

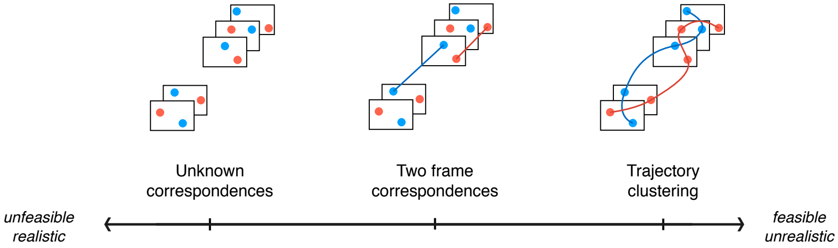


http://www.vision.ee.ethz.ch/~dragonr/airport

https://www.ece.nus.edu.sg/stfpage/eleclf/MTPV_data.zip

http://www.vision.jhu.edu/data/hopkins155/

Experiments - Comments



- Methods belonging to a specific category are **sub-optimal** when applied to the task associated to another category.
- The **local solution** (Mode) is more accurate than the spectral solution (Synch) for motion segmentation with two-frame correspondences.

Outline

1. Synchronization & Motion Segmentation
   - Introduction: synchronization, spectral methods and applications
   - Motion segmentation: spectral method
   - Motion segmentation: local method
   - Concluding remarks

2. Cycle consistency & Localization
   - Introduction: translation synchronization and localization problems
   - Bearing-based localization and parallel rigidity
   - Application to structure from motion
   - Concluding remarks

Summary

Spectral solution

- ✓ It is very **general**: it can be applied to any group admitting a matrix representation (rotations/rigid-motions/permutations/homographies) and also to sets that do not have the structure of a group (binary matrices).
- ✓ It involves a variety of **applications** in Computer Vision (structure from motion, registration, matching, mosaicking, motion segmentation, ...).
- ✓ It entails a **simple** implementation: spectral decomposition + projection.
- ✗ It is an **approximate** solution that does not enforce geometric constraints.

Local solution

- ✓ It is more **accurate** than the spectral solution on motion segmentation datasets.
- ✓ The framework is **modular**: it can be easily generalized to more practical scenarios (e.g. the case of unknown number of motions).
- ✗ The algorithm is **specific** for the motion segmentation problem.

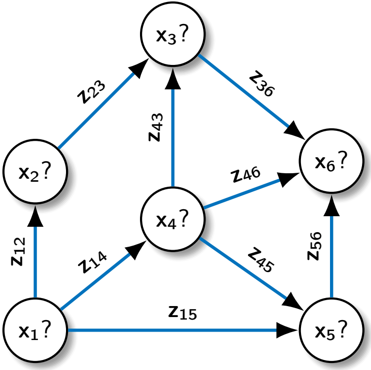
Outline

1. Synchronization & Motion Segmentation
   - Introduction: synchronization, spectral methods and applications
   - Motion segmentation: spectral method
   - Motion segmentation: local method
   - Concluding remarks

2. Cycle consistency & Localization
   - Introduction: translation synchronization and localization problems
   - Bearing-based localization and parallel rigidity
   - Application to structure from motion
   - Concluding remarks

Translation synchronization

Let us consider the **translation synchronization** problem: the task is to recover the position of n nodes in d-space, starting from pairwise differences.



Consistency Constraint

$$\boxed{\mathbf{z}_{ij} = \mathbf{x}_i - \mathbf{x}_j}$$

$\mathbf{x}_i \in \mathbb{R}^d$ unknown

$\mathbf{z}_{ij} \in \mathbb{R}^d$ known

The solution is defined up to a global translation.

Translation synchronization

The unknown node locations can be recovered as the solution of a **linear system** of equations.

$$\mathbf{x}_i - \mathbf{x}_j = \mathbf{z}_{ij} \quad \forall\,(i,j) \in \mathcal{E} \quad \Longleftrightarrow \quad XB = Z \quad \Longleftrightarrow \quad \boxed{(B^{\mathsf{T}} \otimes I_d)\,\mathrm{vec}(X) = \mathrm{vec}(Z)}$$

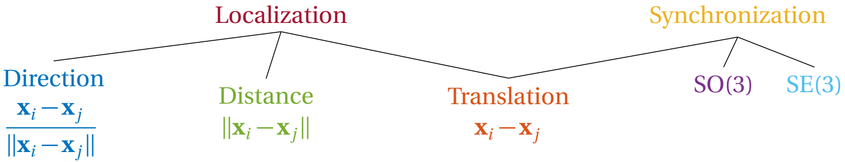
$$\begin{aligned} B &= \text{ incidence matrix} \\ \otimes &= \text{ Kronecker product} \\ I_d &= d \times d \text{ identity matrix} \\ X &= [\mathbf{x}_1 \dots \mathbf{x}_n] \\ Z &= [\mathbf{z}_{12} \ \dots \ \mathbf{z}_{ij} \ \dots] \end{aligned}$$

Remark. Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ connected $\Longrightarrow \mathrm{rank}(B) = n-1 \Longrightarrow \mathrm{rank}(B^{\mathsf{T}} \otimes I_d) = dn - d$.
The rank deficiency corresponds to the **translation ambiguity**.

W. Russel, D. Klein and J. Hespanha Optimal estimation on the graph cycle space. IEEE Transactions on Signal Processing (2011)

Localization

The goal of localization is to compute the position of n nodes in d-space given measures (directions/distances/differences) on the edges.



We are interested here in **direction-based** localization (a.k.a. bearing-based localization), which is **not** a synchronization problem.

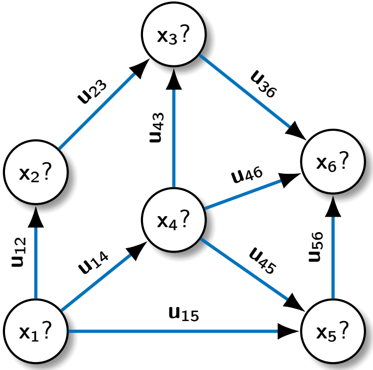
Outline

1. Synchronization & Motion Segmentation
   - Introduction: synchronization, spectral methods and applications
   - Motion segmentation: spectral method
   - Motion segmentation: local method
   - Concluding remarks

2. Cycle consistency & Localization
   - Introduction: translation synchronization and localization problems
   - Bearing-based localization and parallel rigidity
   - Application to structure from motion
   - Concluding remarks

Bearing-based localization

The goal is to recover the position of n nodes in d-space, where pairs of nodes can measure the **direction** of the line joining their locations.



Direction Constraint

$$\mathbf{u}_{ij} = \frac{\mathbf{x}_i - \mathbf{x}_j}{||\mathbf{x}_i - \mathbf{x}_j||}$$

The solution is defined up to **translation and scale**.

Bearing-based localization

Let us rewrite the consistency constraint of translation synchronization in terms of **directions** (known) and **magnitudes** (unknown).

$$\mathbf{x}_i - \mathbf{x}_j = \mathbf{z}_{ij} \quad \forall\, (i,j) \in \mathcal{E} \quad \Longleftrightarrow \quad \boxed{(B^\mathsf{T} \otimes I_d)\,\mathrm{vec}(X) = \mathrm{vec}(Z)}$$

$$\mathbf{x}_i - \mathbf{x}_j = \alpha_{ij}\mathbf{u}_{ij} \quad \forall\, (i,j) \in \mathcal{E} \quad \Longleftrightarrow \quad \boxed{(B^\mathsf{T} \otimes I_d)\,\mathrm{vec}(X) = (I_m \odot U)\boldsymbol{\alpha}}$$

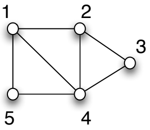
$$m = \text{number of edges}$$
$$\odot = \text{Khatri-Rao product}$$
$$U = [\mathbf{u}_{12} \ \ \ldots \ \ \mathbf{u}_{ij} \ \ \ldots]$$
$$\boldsymbol{\alpha} = [\alpha_{12} \ \ \ldots \ \ \alpha_{ij} \ \ \ldots]^\mathsf{T}$$

Let us consider a **cycle matrix** C and multiply both sides by $C \otimes I_d$.

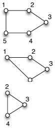
$$(CB^\mathsf{T} \otimes I_d)\,\mathrm{vec}(X) = (C \odot U)\boldsymbol{\alpha} \quad \underset{CB^\mathsf{T}=0}{\Longrightarrow} \quad \boxed{0 = (C \odot U)\boldsymbol{\alpha}}$$

Bearing-based localization

Given a graph we can compute a **cycle basis**, represented as a cycle matrix.



$$
\begin{array}{c}
\begin{array}{ccccccc} (1,2) & (2,4) & (4,1) & (2,3) & (3,4) & (4,5) & (5,1) \end{array} \\
C = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}
\end{array}
$$



The equation $(C \odot U)\alpha = 0$ means that we are imposing **cycle-consistency**: *the sum of differences along any cycle in a basis is zero.*

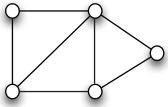
$$
(C \odot U)\alpha = 0 \iff \begin{bmatrix} \mathbf{u}_{12} & \mathbf{0} & \mathbf{0} & \mathbf{u}_{23} & \mathbf{u}_{34} & \mathbf{u}_{45} & \mathbf{u}_{51} \\ \mathbf{u}_{12} & \mathbf{0} & \mathbf{u}_{41} & \mathbf{u}_{23} & \mathbf{u}_{34} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{u}_{24} & \mathbf{0} & \mathbf{u}_{23} & \mathbf{u}_{34} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \alpha_{12} \\ \alpha_{24} \\ \alpha_{41} \\ \alpha_{23} \\ \alpha_{34} \\ \alpha_{45} \\ \alpha_{51} \end{bmatrix} = \mathbf{0}
$$

Bearing-based localizability

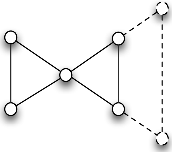
Requiring that the underlying graph is **connected** is **not** sufficient for unique localizability.

The graph is required to be **parallel rigid**: *all the configurations with parallel edges differ by translation and scale*. Otherwise it is called flexible.

Parallel Rigid



Flexible



Remark. If we assume **generic** configurations, then parallel rigidity depends on the graph and dimension d only.

W. Whiteley Matroids from Discrete Geometry *American Mathematical Society* (1997)

T. Eren and W. Whiteley and P. N. Belhumeur Using angle of arrival (bearing) information in network localization *IEEE Conference on Decision and Control* (2006)

Bearing-based localizability

Proposition. ***Node locations*** *can be uniquely (up to translation and scale) determined from pairwise directions if and only if* ***edge lengths*** *can be uniquely (up to scale) recovered from pairwise directions.*

Corollary. *A graph is parallel rigid in d-space if and only if the equation* $(C \odot U)\alpha = 0$ *admits a unique (up to scale) solution, or, equivalently, if and only if* $\boxed{\operatorname{rank}(C \odot U) = m - 1}$ *where*

$$C = \text{cycle matrix}$$

$$\odot = \text{Khatri-Rao product}$$

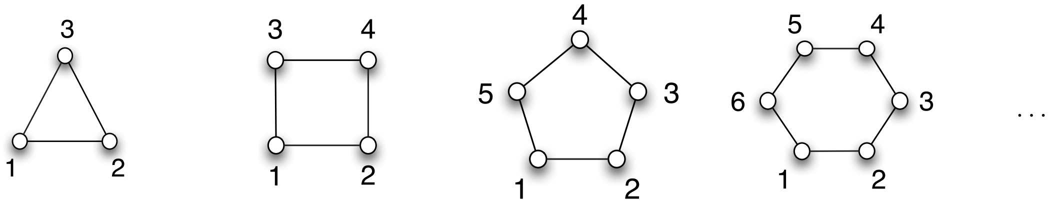
$$U = [\mathbf{u}_{12} \ \ \ldots \ \ \mathbf{u}_{ij} \ \ \ldots]$$

$$m = \text{number of edges}$$

F. Arrigoni and A. Fusiello Bearing-based network localizability: a unifying view IEEE Transactions on Pattern Analysis and Machine Intelligence (2019)

Bearing-based localizability

Proposition. *A cycle of length ℓ is parallel rigid in d-space if and only if* $\boxed{\ell \le d+1}$.



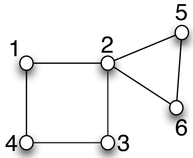
parallel rigid | parallel rigid | flexible | flexible

F. Arrigoni, A. Fusiello and B. Rossi On computing the translations norm in the epipolar graph 3DV (2015)

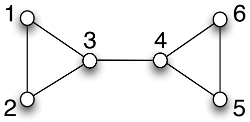
R. Tron, L. Carlone, F. Dellaert, and K. Daniilidis Rigid components identification and rigidity enforcement in bearing-only localization using the graph cycle basis IEEE American Control Conference (2015)

Bearing-based localizability

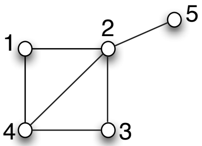
Proposition. *If a graph is parallel rigid in d-space the it does **not** have neither **articulation points** nor **bridges** (i.e., it remains connected after removing any node/edge).*



flexible



flexible

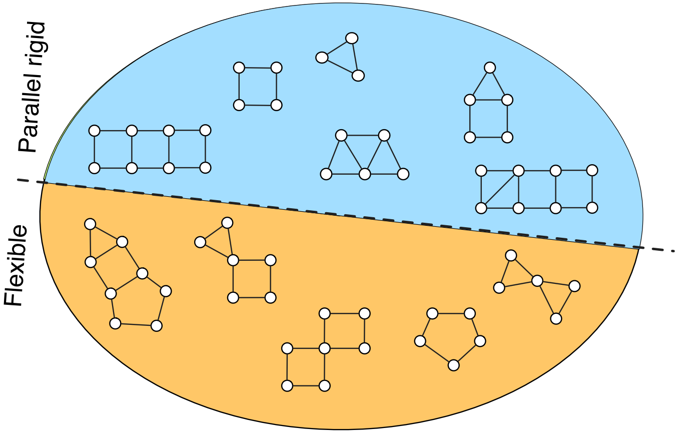


flexible

F. Arrigoni, A. Fusiello and B. Rossi On computing the translations norm in the epipolar graph 3DV (2015)

Bearing-based localizability

Remark: the union of rigid graphs with (at least) one edge in common is also rigid.



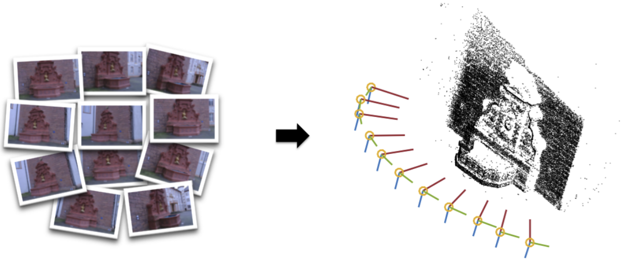
Outline

1. Synchronization & Motion Segmentation
   - Introduction: synchronization, spectral methods and applications
   - Motion segmentation: spectral method
   - Motion segmentation: local method
   - Concluding remarks

2. Cycle consistency & Localization
   - Introduction: translation synchronization and localization problems
   - Bearing-based localization and parallel rigidity
   - Application to structure from motion
   - Concluding remarks

Application to Structure from Motion

The task is to recover both camera positions/orientations and the coordinates of 3D points, starting from multiple images of a static scene.

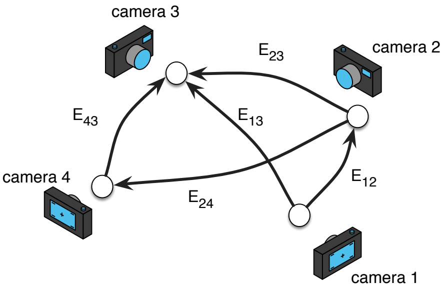


The magnitudes of pairwise translations are unknown and only **directions** can be computed: estimating **camera positions** is an instance of bearing-based localization in 3-space.

O. Ozyesil and A. Singer Robust camera location estimation by convex programming CVPR (2015)

Application to Structure from Motion

Viewing graph: nodes correspond to cameras/images; edges correspond to epipolar geometries.



- If the **viewing graph** is parallel rigid, then the problem is well-posed.
- If the graph is flexible, then the **largest rigid component** has to be extracted.

R. Kennedy, K. Daniilidis, O. Naroditsky, and C. J. Taylor Identifying maximal rigid components in bearing-based localization, Proc. Int. Conf. Intell. Robots Syst. (2012)

Application to Structure from Motion

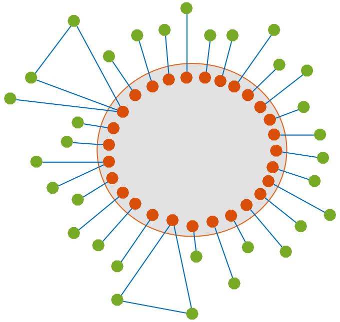
Internet Photo Collections

Dataset	nodes	% edges	rigid	articulation	bridges
Arts Quad	5530	2	✗	30	10
Piccadilly	2508	10	✗	59	62
Roman Forum	1134	11	✗	28	28
Union Square	930	6	✗	60	68
Vienna Cathedral	918	25	✗	19	20
Alamo	627	50	✗	17	19
Notre Dame	553	68	✓	–	–
Tower of London	508	19	✗	19	19
Montreal N. Dame	474	47	✗	7	7
Yorkminster	458	26	✗	9	10
Madrid Metropolis	394	31	✗	17	15
NYC Library	376	29	✗	17	18
Piazza del Popolo	354	40	✗	8	9
Ellis Island	247	67	✗	6	7

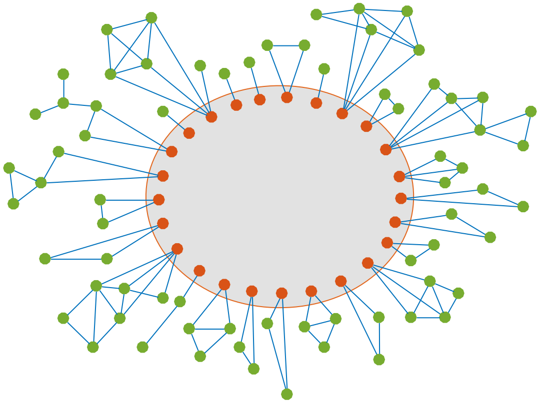
https://research.cs.cornell.edu/1dsfm/

Application to Structure from Motion

Simplified representation: only those edges outside the largest rigid component are drawn.



Roman Forum



Arts Quad

Outline

1. Synchronization & Motion Segmentation
   - Introduction: synchronization, spectral methods and applications
   - Motion segmentation: spectral method
   - Motion segmentation: local method
   - Concluding remarks

2. Cycle consistency & Localization
   - Introduction: translation synchronization and localization problems
   - Bearing-based localization and parallel rigidity
   - Application to structure from motion
   - Concluding remarks

Summary

- Translation synchronization and bearing-based localization are localization problems with different input measures (pairwise differences vesus directions):
  - translation synchronization is well-posed with a **connected** graph;
  - bearing-based localization is well-posed with a **parallel-rigid** graph.

- Bearing-based localizability can be formulated in terms of **cycle-consistency**: this gives an intuitive way to look at rigidity since the problem is expressed in terms of cycles.

- **Structure from motion** is an instance of bearing-based localization in 3-space. Most popular datasets are not rigid (they contain bridges and articulation points).

- Parallel rigidity assumes a **calibrated** setting within structure from motion. If cameras are uncalibrated then a more general notion of solvability is required.

  M. Trager, B. Osserman and J. Ponce On the solvability of viewing graphs ECCV (2018)

More details at

https://synchinvision.github.io/

Appendix

1. Kronecker product
2. Khatri-Rao product
3. Cycle bases
4. Matrices associated with graphs

Kronecker Product

Let A and B be two matrices of dimension $m \times r$ and $n \times s$ respectively.

$$A \otimes B = \begin{bmatrix} [A]_{1,1}B & [A]_{1,2}B & \dots & [A]_{1,r}B \\ [A]_{2,1}B & [A]_{2,2}B & \dots & [A]_{2,r}B \\ \dots & & & \dots \\ [A]_{m,1}B & [A]_{m,2}B & \dots & [A]_{m,r}B \end{bmatrix}$$

- $A \otimes B$ has dimension $mn \times rs$.
- $\otimes$ is associative, distributive, but not commutative.

$$(A \otimes B)^{\mathsf{T}} = A^{\mathsf{T}} \otimes B^{\mathsf{T}}$$

$$(A \otimes B)^{-1} = A^{-1} \otimes B^{-1}$$

$$(A \otimes B)(C \otimes D) = (AC) \otimes (BD)$$

$$\mathrm{vec}(AXB) = (B^{\mathsf{T}} \otimes A)\mathrm{vec}(X)$$

$$\mathrm{rank}(A \otimes B) = \mathrm{rank}(A)\ \mathrm{rank}(B)$$

vec$(\cdot)$ transforms a matrix into a vector by stacking the columns one under the other.

Khatri-Rao Product

Consider two matrices A and B of dimension $m \times r$ and $n \times r$ respectively, and denote the columns of A by $\mathbf{a}_1, \ldots, \mathbf{a}_r$ and those of B by $\mathbf{b}_1, \ldots, \mathbf{b}_r$.

$$A \odot B = \begin{bmatrix} \mathbf{a}_1 \otimes \mathbf{b}_1 & \mathbf{a}_2 \otimes \mathbf{b}_2 & \cdots & \mathbf{a}_r \otimes \mathbf{b}_r \end{bmatrix}$$

- $A \odot B$ has dimension $mn \times r$.
- $\odot$ is associative, distributive, but not commutative.

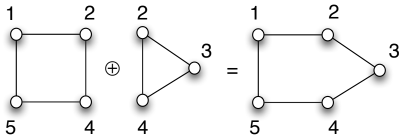
$$(A \otimes B)(C \odot D) = (AC) \odot (BD)$$
$$\mathrm{vec}(A\,\mathrm{diag}(\mathbf{x})B) = (B^\top \odot A)\mathbf{x}$$

$\mathrm{diag}(\mathbf{x})$ transforms the vector $\mathbf{x} = [x_1 \;\ldots\; x_r]^\top$ into a diagonal matrix with elements $x_1, \ldots, x_r$ along the diagonal.

Cycle Bases

A **cycle** in an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is a subgraph in which every vertex has even degree. A **circuit** is a connected cycle where every vertex has degree two.

The **sum** of two cycles is a cycle where the common edges vanish. Viewing cycles as vectors indexed by edges, addition of cycles corresponds to modulo-2 sum of vectors.

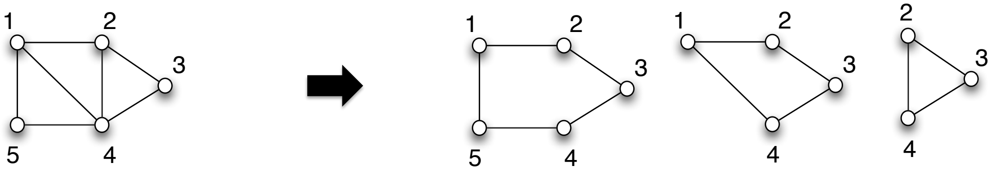


$$\begin{bmatrix}1\\0\\0\\1\\1\\1\end{bmatrix} + \begin{bmatrix}0\\1\\1\\1\\0\\0\end{bmatrix} = \begin{bmatrix}1\\1\\1\\0\\1\\1\end{bmatrix} \quad \begin{array}{c}(1,2)\\(2,3)\\(3,4)\\(2,4)\\(4,5)\\(5,1)\end{array}$$

Cycle Bases

A **cycle basis** is a minimal set of circuits such that any cycle can be written as linear combination of the circuits in the basis.

The dimension of such a space is $m - n + c$, where c denotes the number of connected components in $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, n is the number of vertices and m is the number of edges.



In a directed graph, cycles are represented by **signed** indicator vectors, where the sign indicates whether the orientation of the cycle is concordant with the edge orientation or not.

If we stack the indicator vectors of the circuits of a basis in a matrix C (by rows) we obtain the **cycle matrix**.

Matrices Associated with Graphs

Let us consider a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $n = |\mathcal{V}|$ and $m = |\mathcal{E}|$.

The **adjacency matrix** A is the $n \times n$ matrix defined by

$$[A]_{i,j} = \begin{cases} 1 & \text{if } (i,j) \in \mathcal{E} \\ 0 & \text{otherwise.} \end{cases}$$

The **degree matrix** D is the $n \times n$ matrix defined by $D = \operatorname{diag}(A\mathbf{1}_{n\times 1})$.

The **incidence matrix** B is the $n \times m$ matrix defined by

$$[B]_{k,e} = \begin{cases} 1 & \text{if } k \text{ is the tail of edge } e, \\ -1 & \text{if } k \text{ is the head of edge } e, \\ 0 & \text{otherwise.} \end{cases}$$

- $\mathcal{G}$ connected $\implies \operatorname{rank}(B) = n - 1$
- $CB^{\mathsf{T}} = 0 \quad \forall$ cycle matrix C