Computer Vision 2 – Lecture 10

Multi-Object Tracking III (06.06.2016)

Prof. Dr. Bastian Leibe, Dr. Jörg Stückler
leibe@vision.rwth-aachen.de, stueckler@vision.rwth-aachen.de

RWTH Aachen University, Computer Vision Group
http://www.vision.rwth-aachen.de

Content of the Lecture

- Single-Object Tracking
- Bayesian Filtering
  - Kalman Filters, EKF
  - Particle Filters
- Multi-Object Tracking
  - Introduction
  - MHT, (JPDAF)
  - Network Flow Optimization
- Visual Odometry
- Visual SLAM & 3D Reconstruction

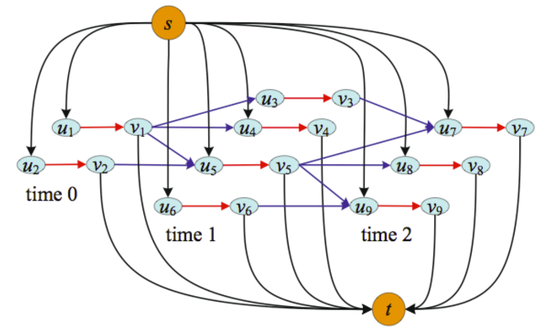


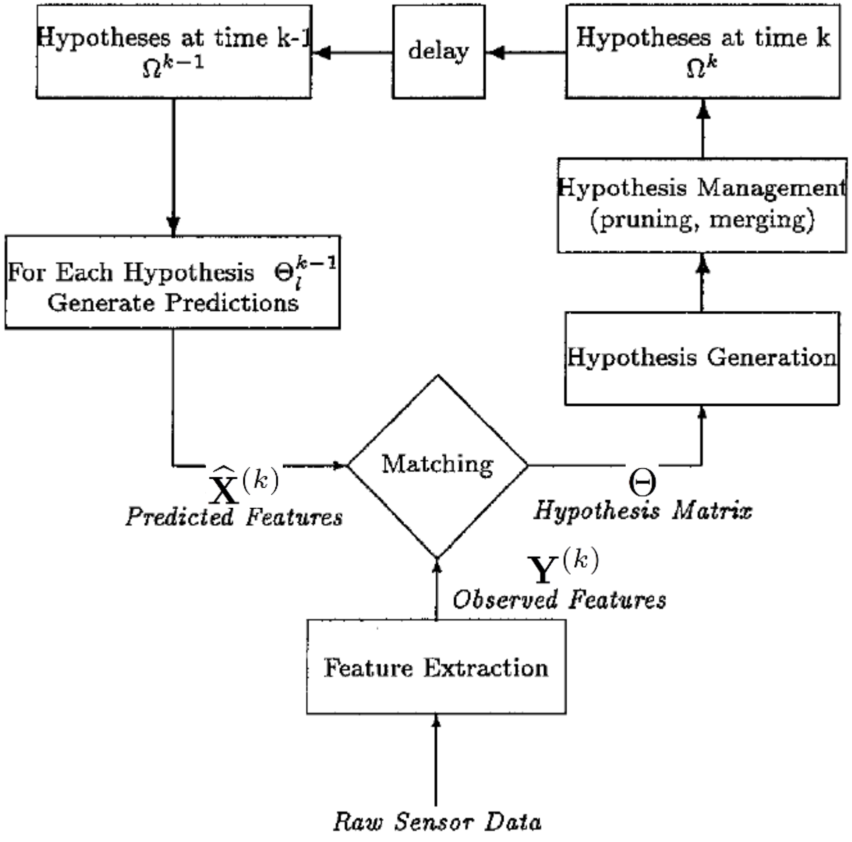
image source: [Zhang, Li, Nevatia, CVPR'08]

Topics of This Lecture

- Recap: MHT
- Data Association as Linear Assignment Problem
  - LAP formulation
  - Greedy algorithm
  - Hungarian algorithm
- Tracking as Network Flow Optimization
  - Min-cost network flow
  - Generalizing to multiple frames
  - Complications
  - Formulation

Recap: Multi-Hypothesis Tracking (MHT)

- Ideas
  - Instead of forming a track tree, keep a set of hypotheses that generate child hypotheses based on the associations.
  - Enforce exclusion constraints between tracks and measure-ments in the assignment.
  - Integrate track generation into the assignment process.
  - After hypothesis generation, merge and prune the current hypothesis set.

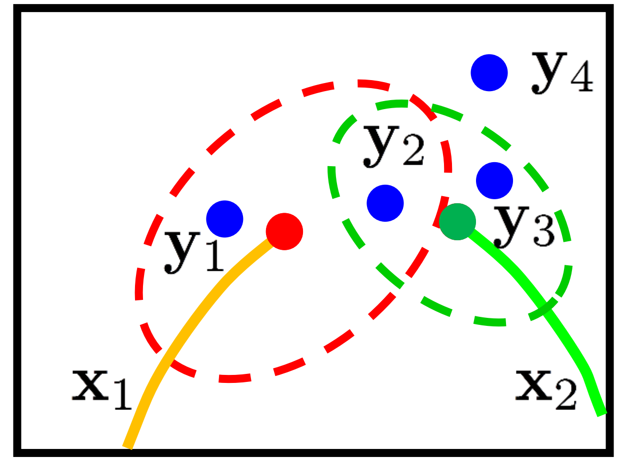


D. Reid, An Algorithm for Tracking Multiple Targets, IEEE Trans. Automatic Control, Vol. 24(6), pp. 843-854, 1979.

Recap: Hypothesis Generation

- Create hypothesis matrix of the feasible associations

$$\Theta = \begin{array}{c} \begin{array}{cccc} \mathbf{x}_1 & \mathbf{x}_2 & \mathbf{x}_{fa} & \mathbf{x}_{nt} \end{array} \\ \begin{bmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \end{array} \begin{array}{c} \\ \mathbf{y}_1 \\ \mathbf{y}_2 \\ \mathbf{y}_3 \\ \mathbf{y}_4 \end{array}$$



- Interpretation
  - Columns represent tracked objects, rows encode measurements
  - A non-zero element at matrix position (i,j) denotes that measurement $\mathbf{y}_i$ is contained in the validation region of track $\mathbf{x}_j$.
  - Extra column $\mathbf{x}_{fa}$ for association as *false alarm*.
  - Extra column $\mathbf{x}_{nt}$ for association as *new track*.
  - Enumerate all *assignments* that are consistent with this matrix.

Recap: Assignments

Z_j	$\mathbf{x}_1$	$\mathbf{x}_2$	$\mathbf{x}_{fa}$	$\mathbf{x}_{nt}$
$\mathbf{y}_1$	0	0	1	0
$\mathbf{y}_2$	1	0	0	0
$\mathbf{y}_3$	0	1	0	0
$\mathbf{y}_4$	0	0	0	1

- Impose constraints
  - A measurement can originate from only one object.

  $\Rightarrow$ Any row has only a single non-zero value.

  - An object can have at most one associated measurement per time step.

  $\Rightarrow$ Any column has only a single non-zero value, except for $\mathbf{x}_{fa}$, $\mathbf{x}_{nt}$

Recap: Calculating Hypothesis Probabilities

- Probabilistic formulation
  - It is straightforward to enumerate all possible assignments.
  - However, we also need to calculate the probability of each child hypothesis.
  - This is done recursively:

$$
\begin{aligned}
p(\Omega_j^{(k)} | \mathbf{Y}^{(k)}) &= p(Z_j^{(k)}, \Omega_{p(j)}^{(k-1)} | \mathbf{Y}^{(k)}) \\
&\overset{Bayes}{=} \eta p(\mathbf{Y}^{(k)} | Z_j^{(k)}, \Omega_{p(j)}^{(k-1)}) p(Z_j^{(k)}, \Omega_{p(j)}^{(k-1)}) \\
&= \underbrace{\eta}_{\text{Normalization factor}} \underbrace{p(\mathbf{Y}^{(k)} | Z_j^{(k)}, \Omega_{p(j)}^{(k-1)})}_{\text{Measurement likelihood}} \underbrace{p(Z_j^{(k)} | \Omega_{p(j)}^{(k-1)})}_{\text{Prob. of assignment set}} \underbrace{p(\Omega_{p(j)}^{(k-1)})}_{\text{Prob. of parent}}
\end{aligned}
$$

Recap: Measurement Likelihood

- Use KF prediction
  - Assume that a measurement $\mathbf{y}_i^{(k)}$ associated to a track $\mathbf{x}_j$ has a Gaussian pdf centered around the measurement prediction $\hat{\mathbf{x}}_j^{(k)}$ with innovation covariance $\widehat{\mathbf{\Sigma}}_j^{(k)}$.
  - Further assume that the pdf of a measurement belonging to a new track or false alarm is uniform in the observation volume W (the sensor's field-of-view) with probability W^{-1}.
  - Thus, the measurement likelihood can be expressed as

$$p\left(\mathbf{Y}^{(k)} | Z_j^{(k)}, \Omega_{p(j)}^{(k-1)}\right) = \prod_{i=1}^{M_k} \mathcal{N}\left(\mathbf{y}_i^{(k)}; \hat{\mathbf{x}}_j, \widehat{\mathbf{\Sigma}}_j^{(k)}\right)^{\delta_i} W^{-(1-\delta_i)}$$

$$= W^{-(N_{fal}+N_{new})} \prod_{i=1}^{M_k} \mathcal{N}\left(\mathbf{y}_i^{(k)}; \hat{\mathbf{x}}_j, \widehat{\mathbf{\Sigma}}_j^{(k)}\right)^{\delta_i}$$

Recap: Probability of an Assignment Set

$$p(Z_j^{(k)} | \Omega_{p(j)}^{(k-1)})$$

- Composed of three terms
  1. Probability of the number of tracks N_{det}, N_{fal}, N_{new}
     - Assumption 1: N_{det} follows a binomial distribution

$$p(N_{det} | \Omega_{p(j)}^{(k-1)}) = \begin{pmatrix} N \\ N_{det} \end{pmatrix} p_{det}^{N_{det}} (1 - p_{det})^{(N - N_{det})}$$

where N is the number of tracks in the parent hypothesis

- Assumption 2: N_{fal} and N_{new} both follow a Poisson distribution with expected number of events $\lambda_{fal} W$ and $\lambda_{new} W$

$$p(N_{det}, N_{fal}, N_{new} | \Omega_{p(j)}^{(k-1)}) = \begin{pmatrix} N \\ N_{det} \end{pmatrix} p_{det}^{N_{det}} (1 - p_{det})^{(N - N_{det})}$$
$$\cdot \mu(N_{fal}; \lambda_{fal} W) \cdot \mu(N_{new}; \lambda_{new} W)$$

Recap: Probability of an Assignment Set

2. Probability of a specific assignment of measurements
   - Such that $M_k = N_{det} + N_{fal} + N_{new}$ holds.
   - This is determined as 1 over the number of combinations

$$\binom{M_k}{N_{det}} \binom{M_k - N_{det}}{N_{fal}} \binom{M_k - N_{det} - N_{fal}}{N_{new}}$$

3. Probability of a specific assignment of tracks
   - Given that a track can be either *detected* or not *detected*.
   - This is determined as 1 over the number of assignments

$$\frac{N!}{(N - N_{det})!} \binom{N - N_{det}}{N_{det}}$$

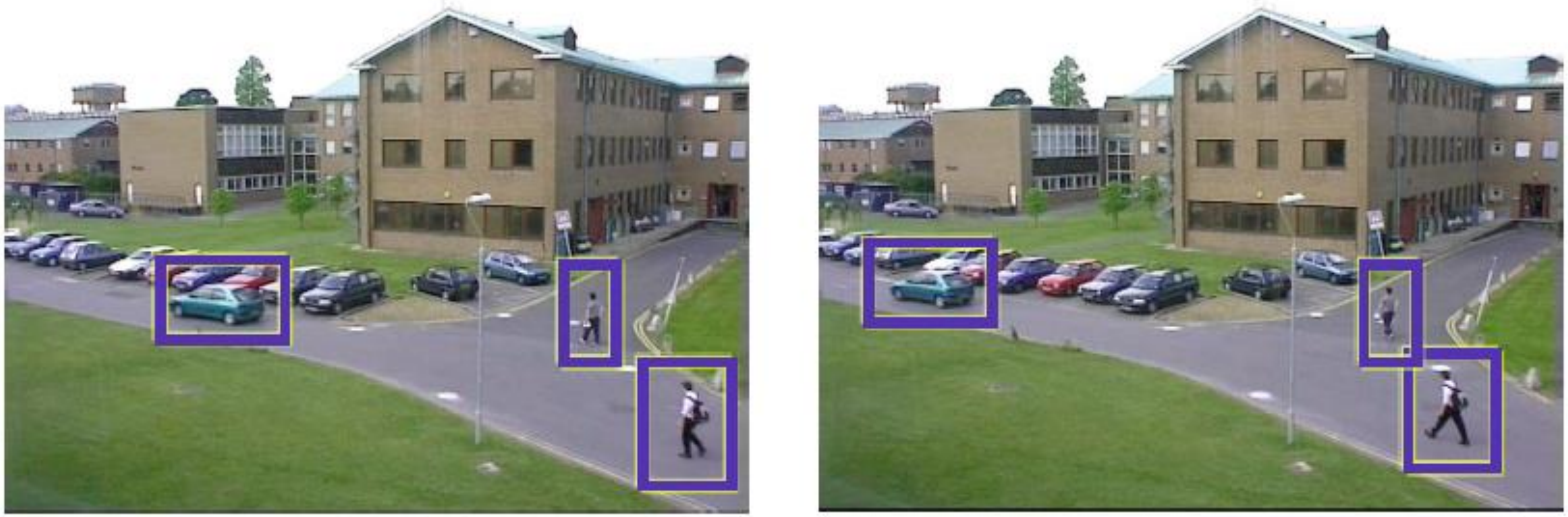
$\Rightarrow$ When combining the different parts, many terms cancel out!

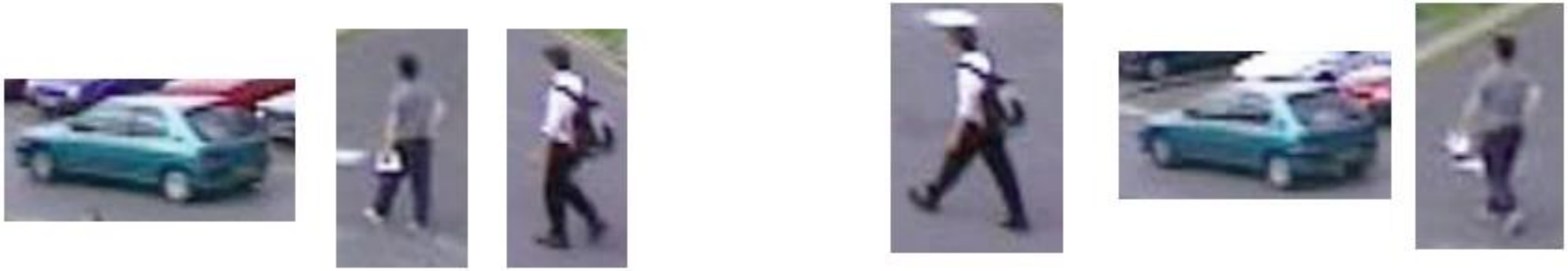
Topics of This Lecture

- Recap: MHT
- Data Association as Linear Assignment Problem
  - LAP formulation
  - Greedy algorithm
  - Hungarian algorithm
- Tracking as Network Flow Optimization
  - Min-cost network flow
  - Generalizing to multiple frames
  - Complications
  - Formulation

Back to Data Association...

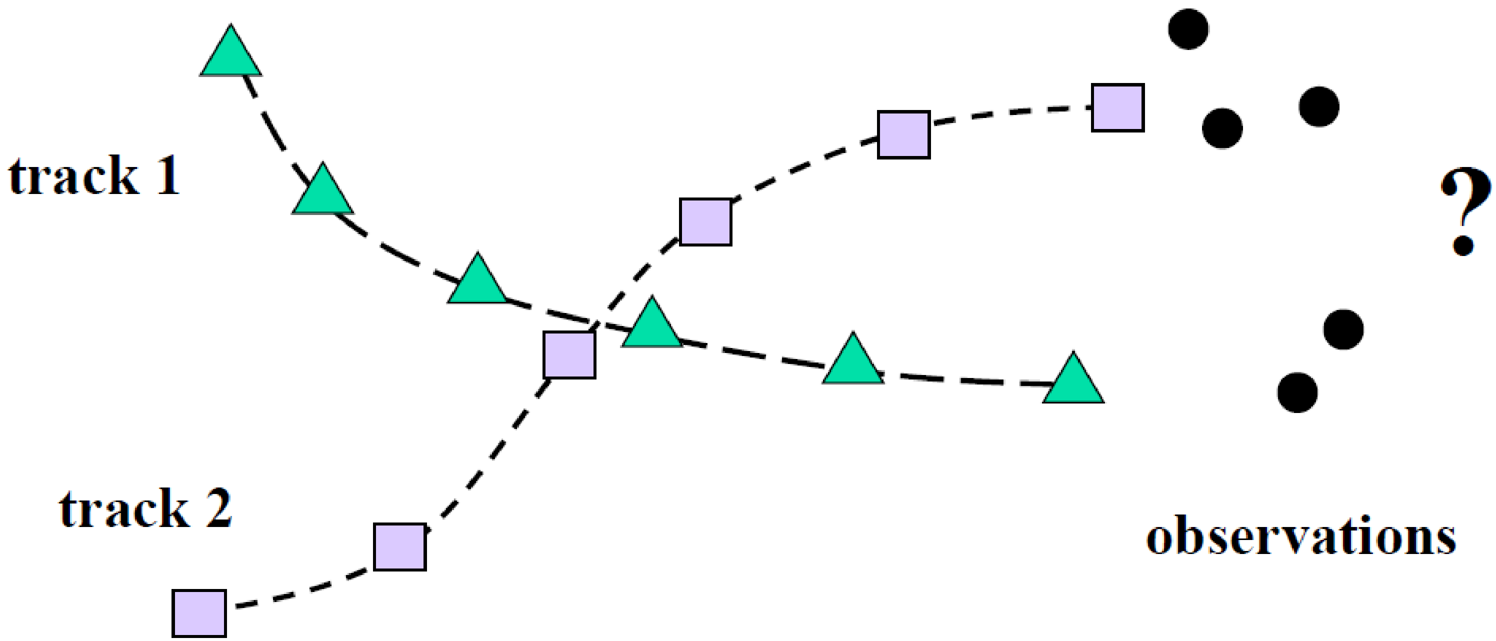
- Goal: Match detections across frames





Slide credit: Robert Collins

Data Association



- Main question here
  - How to determine which measurements to add to which track?
  - Today: consider this as a matching problem

Slide credit: Robert Collins

Linear Assignment Formulation

- Form a matrix of pairwise similarity scores
- Similarity could be
  - based on motion prediction
  - based on appearance
  - based on both

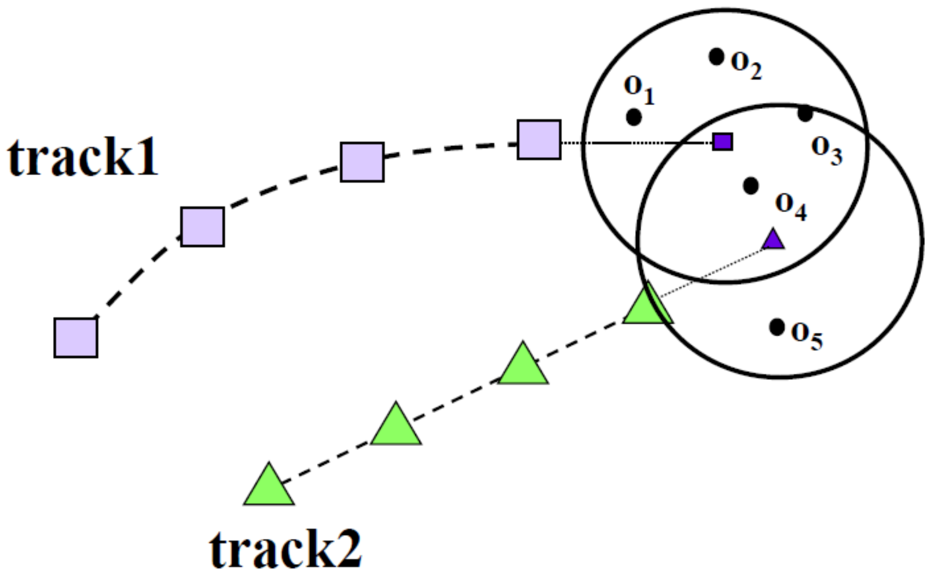
Frame t \ Frame $t+1$			
	0.11	$\boxed{0.95}$	0.23
	0.85	0.25	$\boxed{0.89}$
	$\boxed{0.90}$	0.12	0.81

- Goal
  - Choose one match from each row and column to maximize the sum of scores

Slide credit: Robert Collins

Linear Assignment Formulation

- Example: Similarity based on motion prediction
  - Predict motion for each trajectory and assign scores for each measurement based on inverse (Mahalanobis) distance, such that closer measurements get higher scores.



	ai1	ai2
1	3.0	
2	5.0	
3	6.0	1.0
4	9.0	8.0
5		3.0

  - Choose at most one match in each row and column to maximize sum of scores

Slide credit: Robert Collins

Linear Assignment Problem

- Formal definition
  - Maximize $$\sum_{i=1}^{N} \sum_{j=1}^{M} w_{ij} z_{ij}$$

    subject to $$\left.\begin{aligned} &\textstyle\sum_{j=1} z_{ij} = 1;\ \ i = 1, 2, \ldots, N \\ &\textstyle\sum_{i=1} z_{ij} = 1;\ \ j = 1, 2, \ldots, M \\ &z_{ij} \in \{0, 1\} \end{aligned}\right\}$$

    Those constraints ensure that Z is a permutation matrix

  - The permutation matrix constraint ensures that we can only match up one object from each row and column.
  - Note: Alternatively, we can minimize cost rather than maximizing weights.

$$\arg\min_{z_{ij}} \sum_{i=1}^{N} \sum_{j=1}^{M} c_{ij} z_{ij}$$

Slide credit: Robert Collins

Greedy Solution to LAP

	1	2	3	4	5
1	0.95	0.76	0.62	0.41	0.06
2	0.23	0.46	0.79	0.94	0.35
3	0.61	0.02	0.92	0.92	0.81
4	0.49	0.82	0.74	0.41	0.01
5	0.89	0.44	0.18	0.89	0.14

- Greedy algorithm
  - Find the largest score
  - Remove scores in same row and column from consideration
  - Repeat
- Result: score $=$

Slide credit: Robert Collins

Greedy Solution to LAP

	1	2	3	4	5
1	**0.95**	0.76	0.62	0.41	0.06
2	0.23	0.46	0.79	**0.94**	0.35
3	0.61	0.02	**0.92**	0.92	0.81
4	0.49	**0.82**	0.74	0.41	0.01
5	0.89	0.44	0.18	0.39	**0.14**

- Greedy algorithm
  - Find the largest score
  - Remove scores in same row and column from consideration
  - Repeat
- Result: score $= 0.95 + 0.94 + 0.92 + 0.82 + 0.14 = 3.77$

Is this the best we can do?

Slide credit: Robert Collins

Greedy Solution to LAP

	1	2	3	4	5
1	$\boxed{0.95}$	0.76	0.62	0.41	0.06
2	0.23	0.46	0.79	$\boxed{0.94}$	0.35
3	0.61	0.02	$\boxed{0.92}$	0.92	0.81
4	0.49	$\boxed{0.82}$	0.74	0.41	0.01
5	0.89	0.44	0.18	0.89	$\boxed{0.14}$

Greedy solution
score $= 3.77$

	1	2	3	4	5
1	$\boxed{0.95}$	0.76	0.62	0.41	0.06
2	0.23	0.46	$\boxed{0.79}$	0.94	0.35
3	0.61	0.02	0.92	0.92	$\boxed{0.81}$
4	0.49	$\boxed{0.82}$	0.74	0.41	0.01
5	0.89	0.44	0.18	$\boxed{0.89}$	0.14

Optimal solution
score $= 4.26$

- Discussion
  - Greedy method is easy to program, quick to run, and yields “pretty good” solutions in practice.
  - But it often does not yield the optimal solution.

Slide credit: Robert Collins

Optimal Solution

- Hungarian Algorithm
  - There is an algorithm called Kuhn-Munkres or "Hungarian" algorithm specifically developed to efficiently solve the linear assignment problem.
  - Reduces assignment problem to bipartite graph matching.
  - When starting from an $N \times N$ matrix, it runs in $\mathcal{O}(N^3)$.

  $\Rightarrow$ If you need LAP, you should use this algorithm.

- In the following
  - Look at other algorithms that generalize to multi-frame (>2 frames) problems.

  $\Rightarrow$ Min-Cost Network Flow

Slide credit: Robert Collins

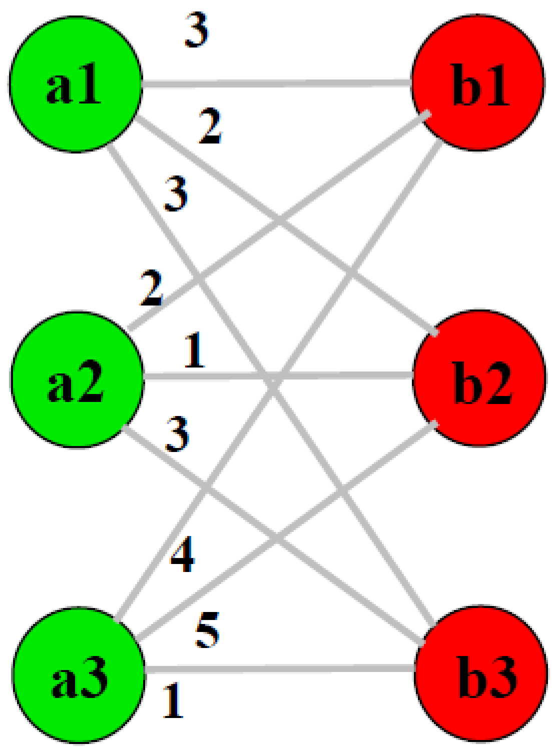
Topics of This Lecture

- Recap: MHT
- Data Association as Linear Assignment Problem
  - LAP formulation
  - Greedy algorithm
  - Hungarian algorithm
- Tracking as Network Flow Optimization
  - Min-cost network flow
  - Generalizing to multiple frames
  - Complications
  - Formulation

Min-Cost Flow

- Small example

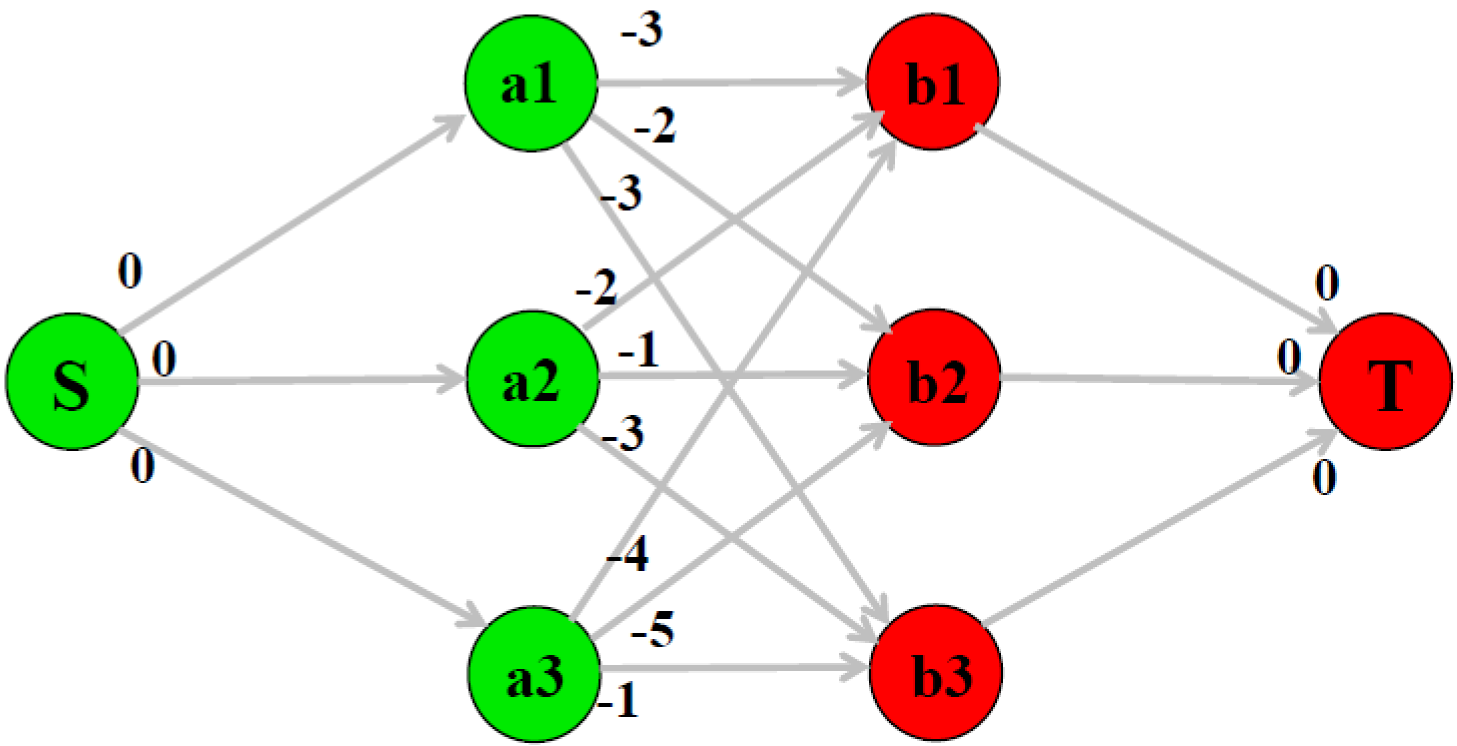
	1	2	3
1	3	2	3
2	2	1	3
3	4	5	1



- Network Flow formulation
  – Reformulate Linear Cost Assignment into a min-cost flow problem

Slide credit: Robert Collins

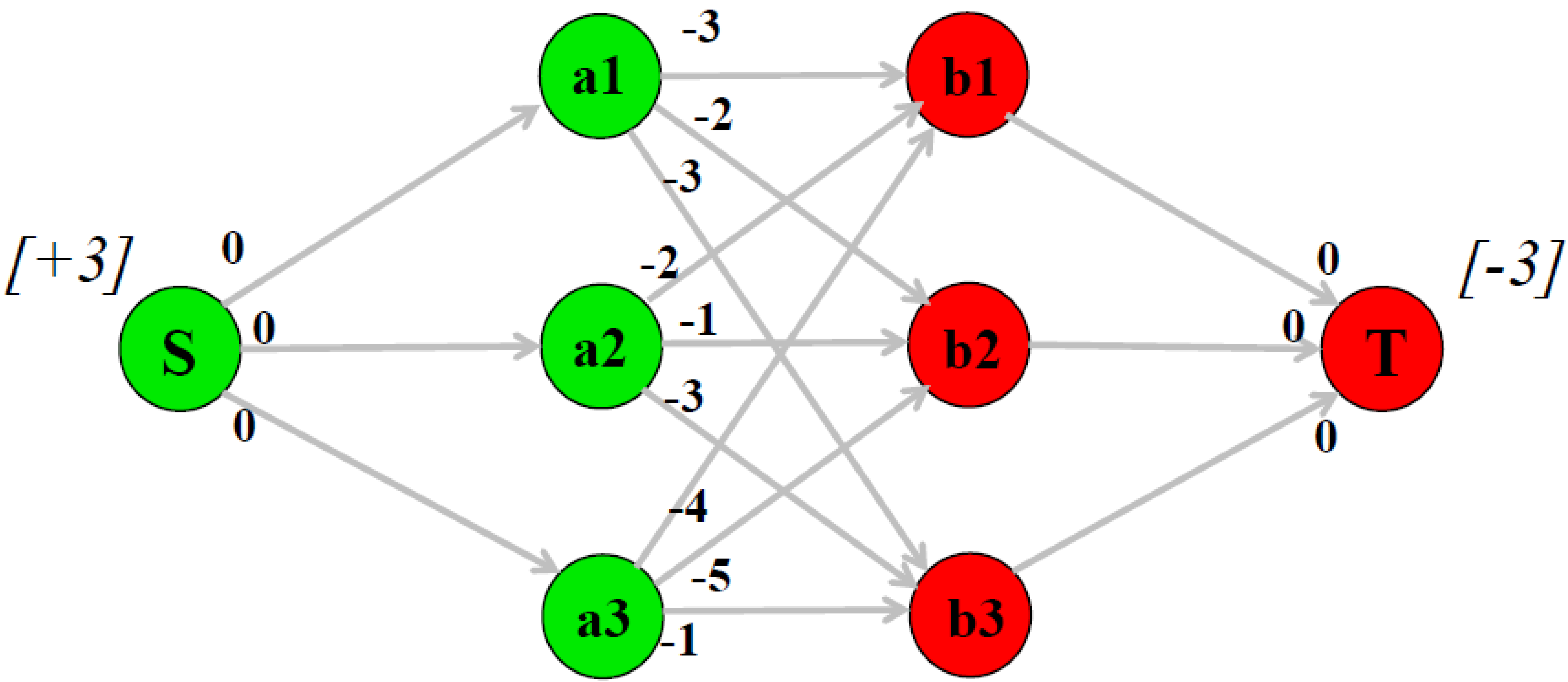
Min-Cost Flow



- Conversion into flow graph
  - Transform weights into costs $c_{ij} = \alpha - w_{ij}$
  - Add source/sink nodes with 0 cost.
  - Directed edges with a capacity of 1.

Slide credit: Robert Collins

Min-Cost Flow

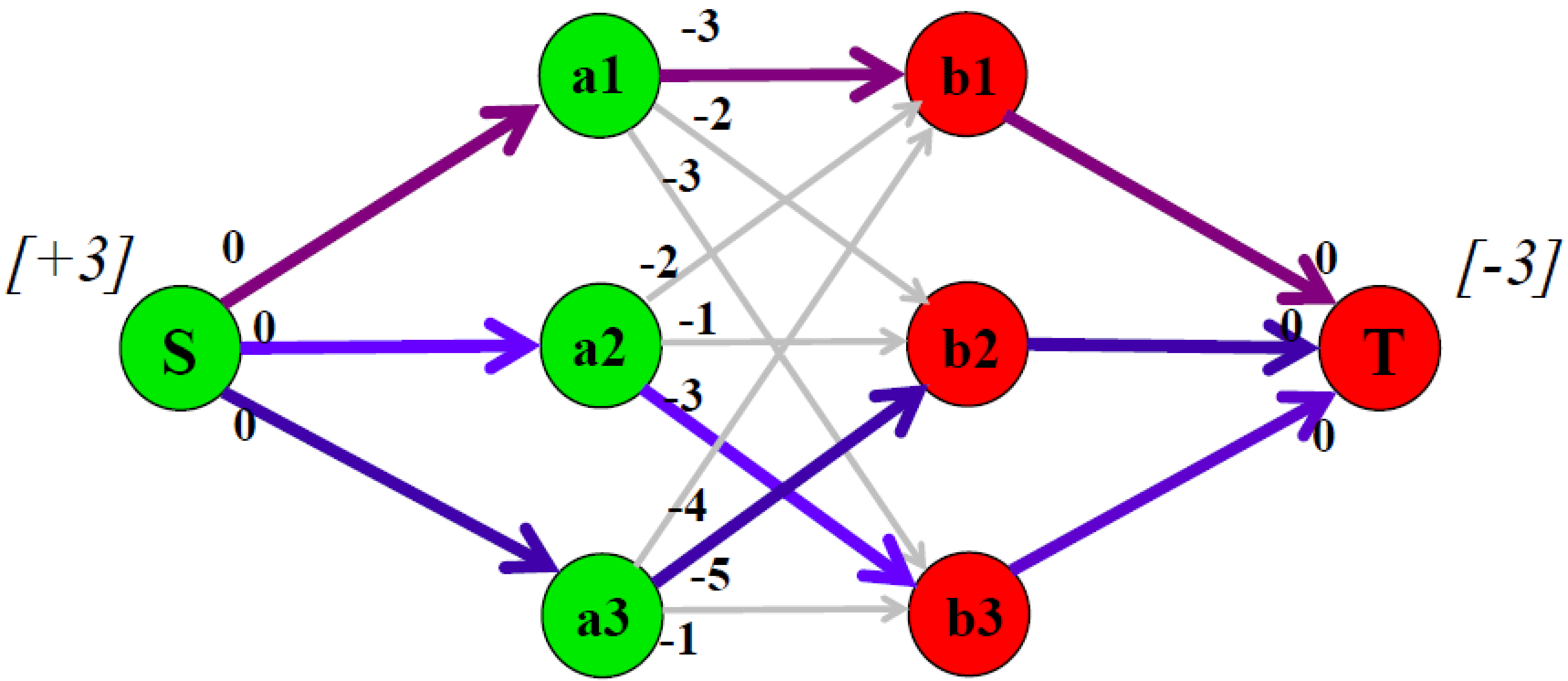


- Conversion into flow graph
  – Pump N units of flow from source to sink.
  – Internal nodes pass on flow ($\sum$ flow in $= \sum$ flow out).

  $\Rightarrow$ Find the optimal paths along which to ship the flow.

Slide credit: Robert Collins

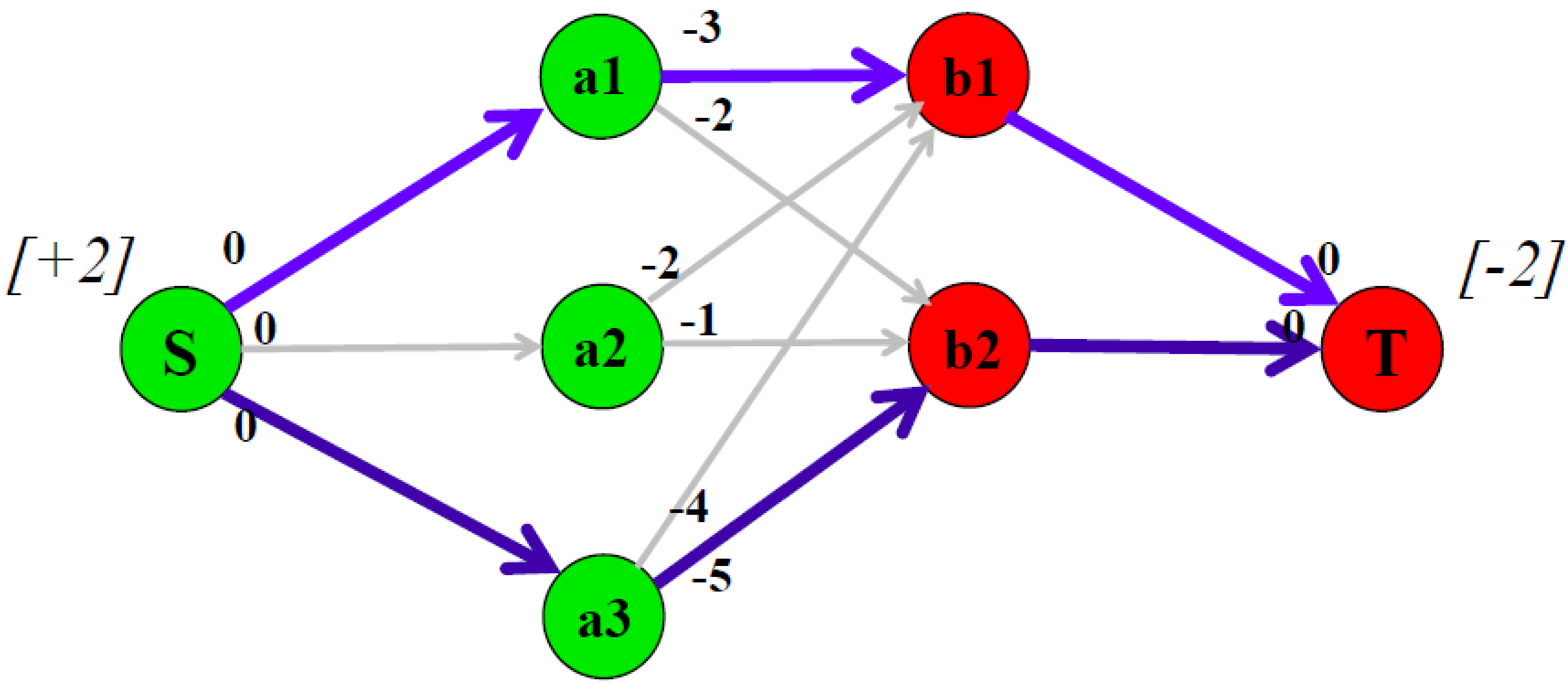
Min-Cost Flow



- Conversion into flow graph
  - Pump N units of flow from source to sink.
  - Internal nodes pass on flow ($\sum$ flow in $=\sum$ flow out).
  - $\Rightarrow$ Find the optimal paths along which to ship the flow.

Slide credit: Robert Collins

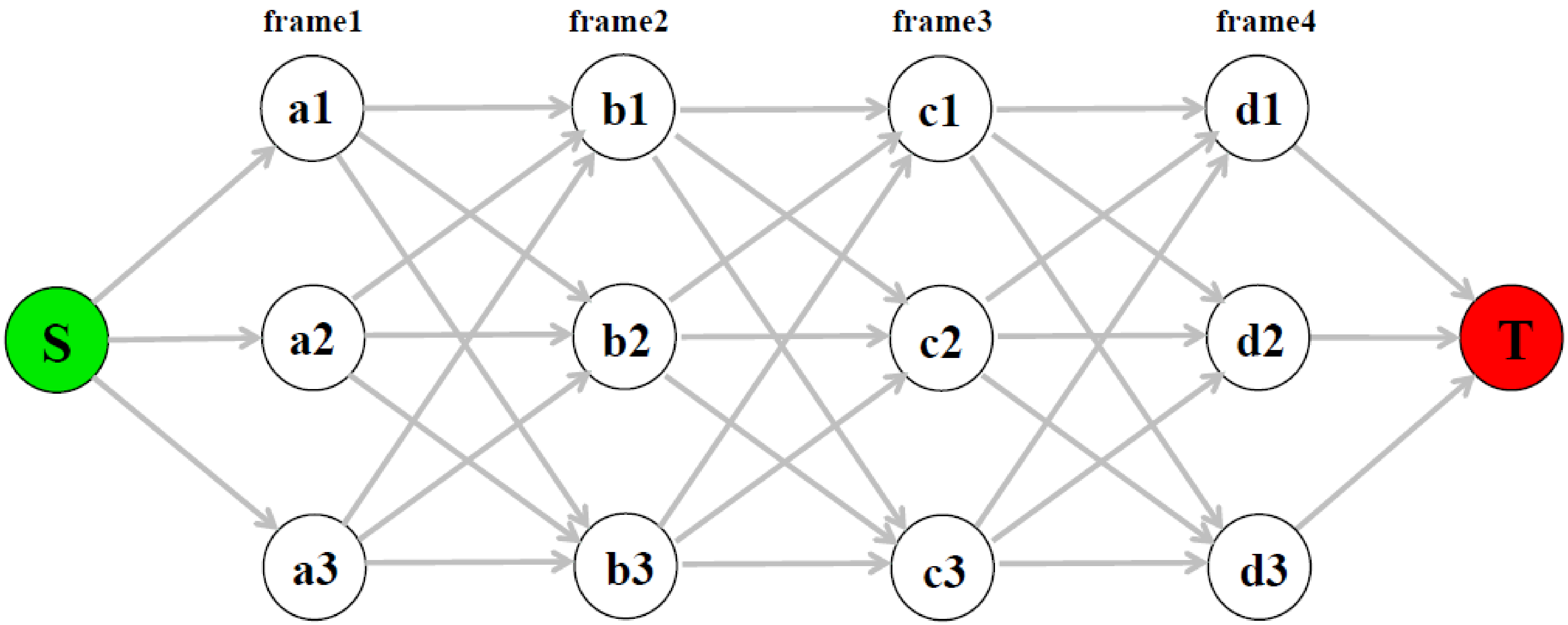
Min-Cost Flow



- Nice property
  - Min-cost formalism readily generalizes to matching sets with unequal sizes.

Slide credit: Robert Collins

Using Network Flow for Tracking

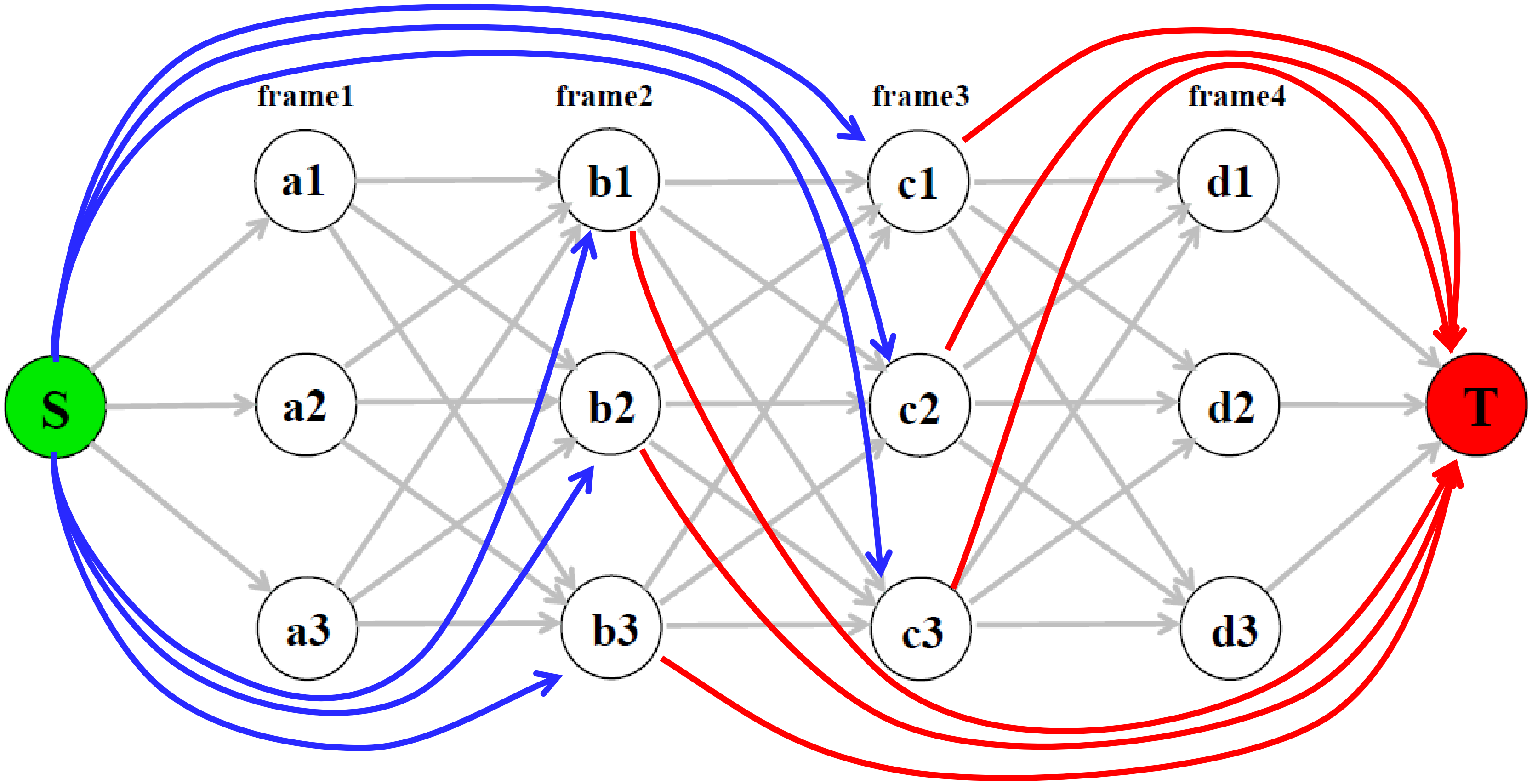


- Approach
  - Seek a globally optimal solution by considering observations over all frames in “batch mode”.

    $\Rightarrow$ Extend two-frame min-cost formulation by adding observations from all frames into the network.

Slide credit: Robert Collins

Using Network Flow for Tracking

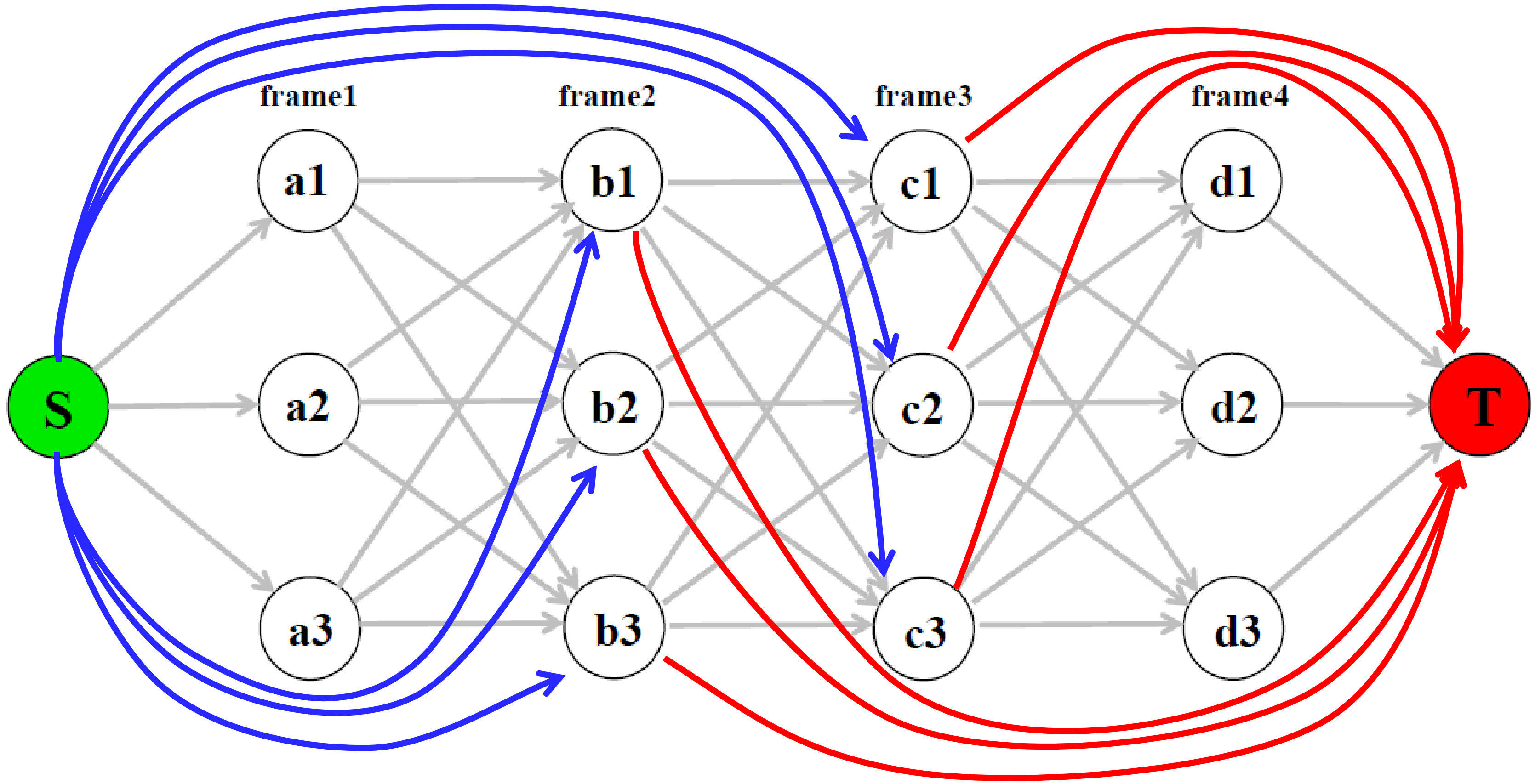


- Complication 1
  - Tracks can start later than frame1 (and end earlier than frame4)

  ⇒ Connect the source and sink nodes to all intermediate nodes.

Slide credit: Robert Collins

Using Network Flow for Tracking

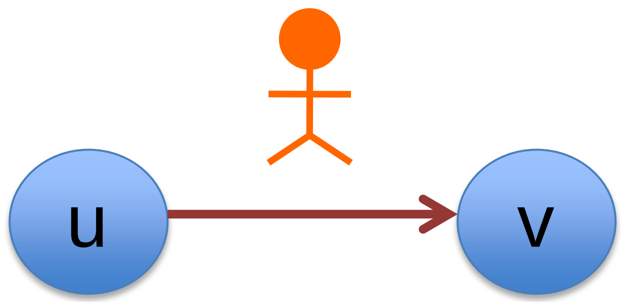


- Complication 2
  - Trivial solution: zero cost flow!

Slide credit: Robert Collins

Using Network Flow for Tracking

- Solution
  - Divide each detection into 2 nodes

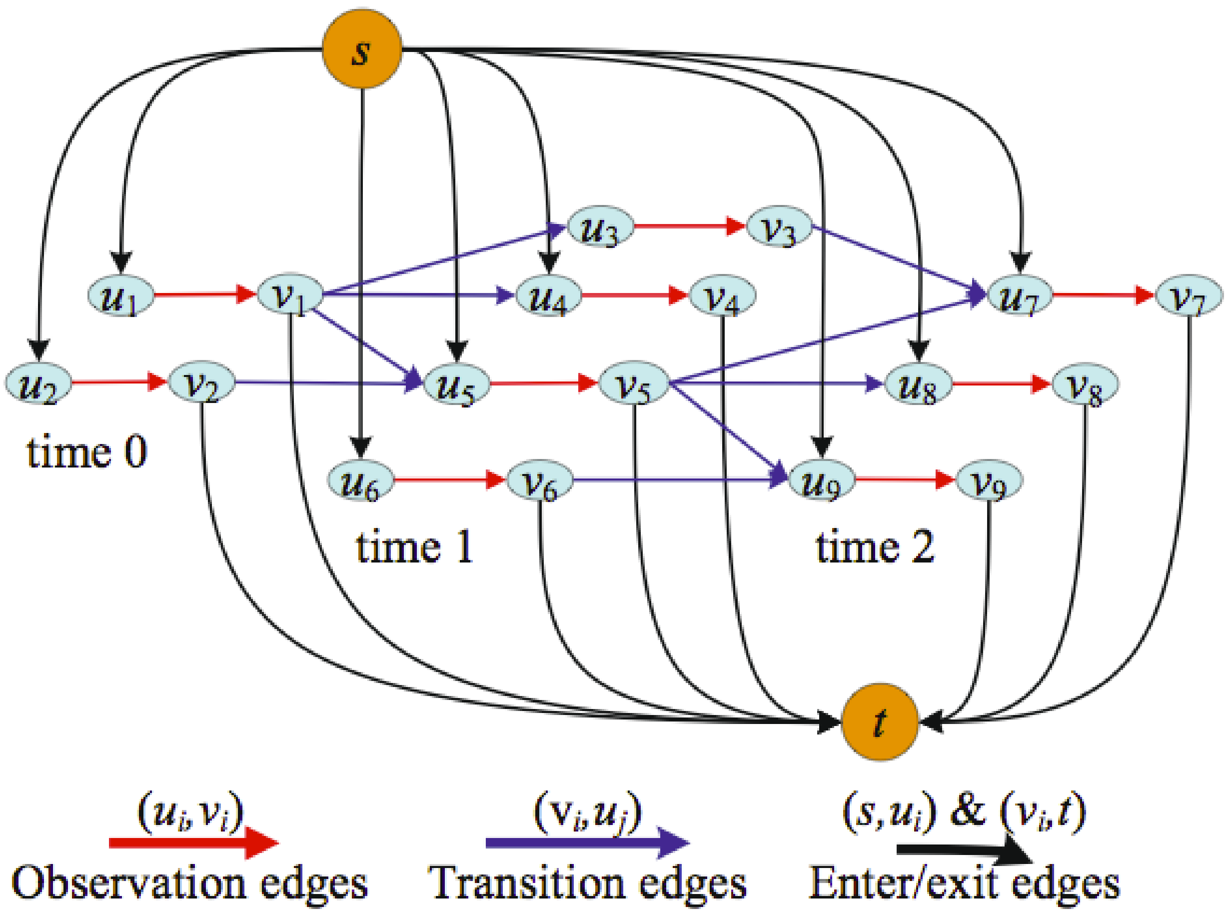


Detection edge

$$C_i = \log \frac{\beta_i}{1-\beta_i} \longleftarrow$$ Probability that detection i is a false alarm

Zhang, Li, Nevatia, [Global Data Association for Multi-Object Tracking using Network Flows](), CVPR'08.

Network Flow Approach



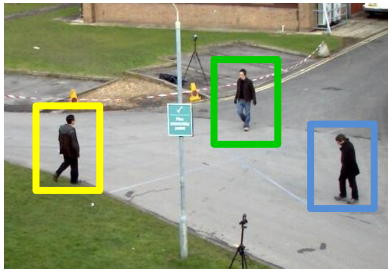
Zhang, Li, Nevatia, Global Data Association for Multi-Object Tracking using Network Flows, CVPR'08.

image source: [Zhang, Li, Nevatia, CVPR'08]

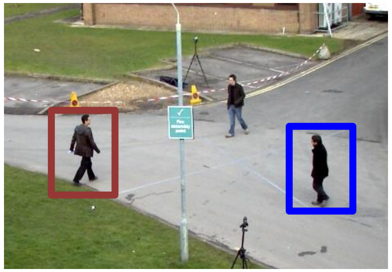
Network Flow Approach: Illustration

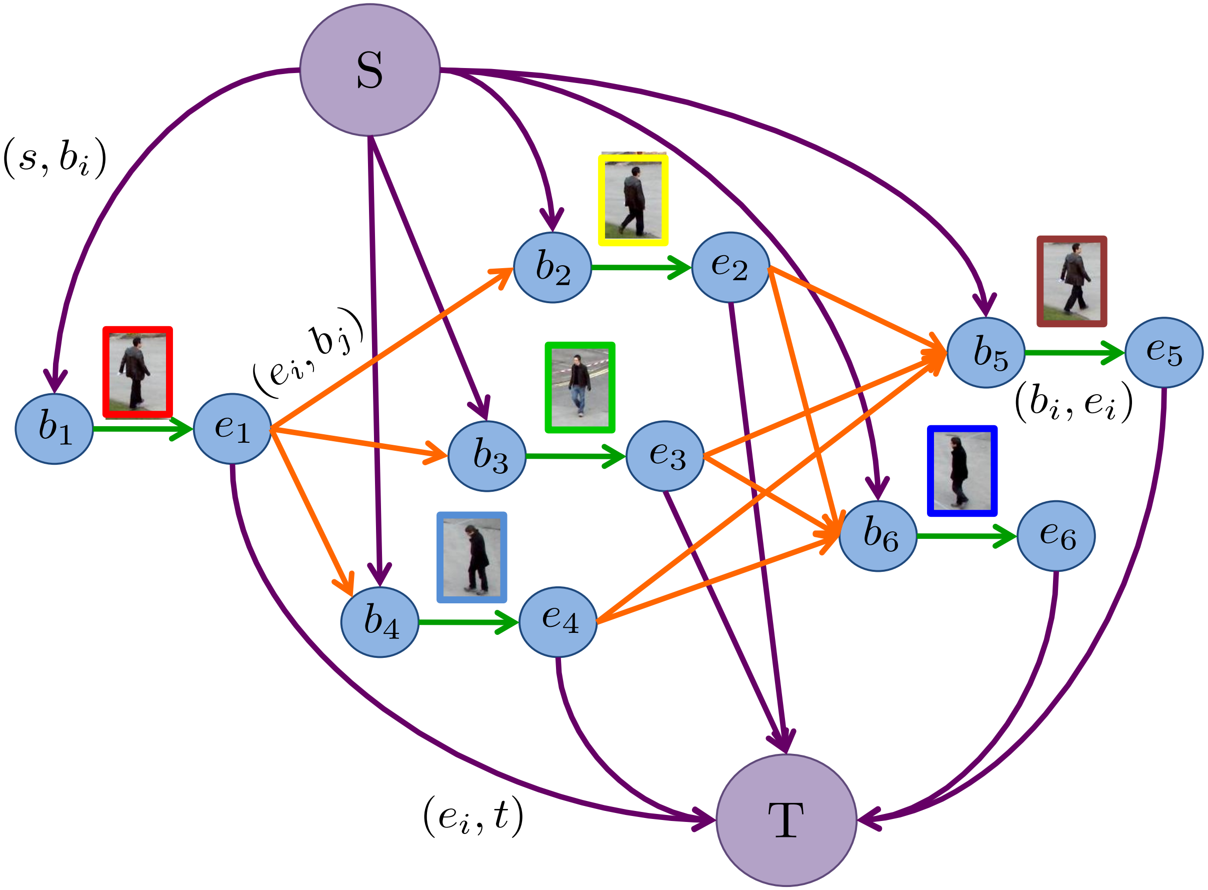
Frame t-1

Frame t



Frame t+1





Slide credit: Laura Leal-Taixe

Min-Cost Formulation

- Objective Function

$$\mathcal{T}* = \underset{\mathcal{T}}{\mathbf{argmin}} \sum_i C_{in,i} f_{in,i} + \sum_i C_{i,out} f_{i,out}$$

$$+ \sum_{i,j} C_{i,j} f_{i,j} + \sum_i C_i f_i$$

- subject to
  - Flow conservation at all nodes

$$f_{in,i} + \sum_j f_{j,i} = f_i = f_{out,i} + \sum_j f_{i,j} \ \forall i$$

  - Edge capacities

$$f_i \leq 1$$

Slide credit: Laura Leal-Taixe

Min-Cost Formulation

- Objective Function

$$\mathcal{T}* = \underset{\mathcal{T}}{\mathbf{argmin}} \sum_i C_{in,i} f_{in,i} + \sum_i C_{i,out} f_{i,out}$$

$$+ \sum_{i,j} C_{i,j} f_{i,j} + \sum_i C_i f_i$$

$$\Big\downarrow \quad C_i = -log(P_i)$$

- Equivalent to Maximum A-Posteriori formulation

$$\mathcal{T}* = \underset{\mathcal{T}}{\mathbf{argmax}} \prod_i P(\mathbf{o}_i|\mathcal{T}) P(\mathcal{T})$$

Slide credit: Laura Leal-Taixe

Min-Cost Formulation

- Objective Function

$$\mathcal{T}* = \underset{\mathcal{T}}{\mathbf{argmin}} \underbrace{\boxed{\sum_i C_{in,i} f_{in,i}}}_{\text{IN}} + \underbrace{\boxed{\sum_i C_{i,out} f_{i,out}}}_{\text{OUT}}$$

$$+ \underbrace{\boxed{\sum_{i,j} C_{i,j} f_{i,j}}}_{\text{TRANSITION}} + \boxed{\sum_i C_i f_i}$$

Likelihood of the detection

- Equivalent to Maximum A-Posteriori formulation

$$\mathcal{T}* = \underset{\mathcal{T}}{\mathbf{argmax}} \boxed{\prod_i P(\mathbf{o}_i | \mathcal{T})} P(\mathcal{T})$$

$$P(\mathcal{T}) = \prod_{T_k \in \mathcal{T}} P(T_k)$$

Independence assumption
+
Markov

Slide credit: Laura Leal-Taixe

Network Flow Solutions

- Push-relabel method
  - Zhang, Li and Nevatia, “**Global Data Association for Multi-Object Tracking Using Network Flows**,” CVPR 2008.
- Successive shortest path algorithm
  - Berclaz, Fleuret, Turetken and Fua, “**Multiple Object Tracking using K-shortest Paths Optimization**,” IEEE PAMI, Sep 2011.
  - Pirsiavash, Ramanan, Fowlkes, “**Globally Optimal Greedy Algorithms for Tracking a Variable Number of Objects**”, CVPR‘11.
  - These both include approximate dynamic programming solutions

Slide credit: Robert Collins

Summary

- Tracking as network flow optimization
- Pros
  - Clear algorithmic framework, equivalence to probabilistic formulation
  - Well-understood LP optimization problem, efficient algorithms available
  - Globally optimal solution
- Cons / Limitations
  - Only applicable to restricted problem setting due to LP formulation
    - Not possible to encode exclusion constraints between detections (e.g., to penalize physical overlap)
    - Motion model can only draw upon information from pairs of detections (i.e., only zero-velocity model possible, no constant velocity models)
  - C_{in} and C_{out} cost terms are quite fiddly to set in practice
    - Too low $\Rightarrow$ fragmentations, too high $\Rightarrow$ ID switches

References and Further Reading

- The original network flow tracking paper
  - Zhang, Li, Nevatia, Global Data Association for Multi-Object Tracking using Network Flows, CVPR'08.
- Extensions and improvements
  - Berclaz, Fleuret, Turetken, Fua, Multiple Object Tracking using K-shortest Paths Optimization, IEEE PAMI, Sep 2011. (code)
  - Pirsiavash, Ramanan, Fowlkes, Globally Optimal Greedy Algorithms for Tracking a Variable Number of Objects, CVPR'11.
- A recent extension to incorporate social walking models
  - L. Leal-Taixe, G. Pons-Moll, B. Rosenhahn, Everybody Needs Somebody: Modeling Social and Grouping Behavior on a Linear Programming Multiple People Tracker, ICCV Workshops 2011.