Globally Optimal Solution to Multi-Object Tracking with Merged Measurements

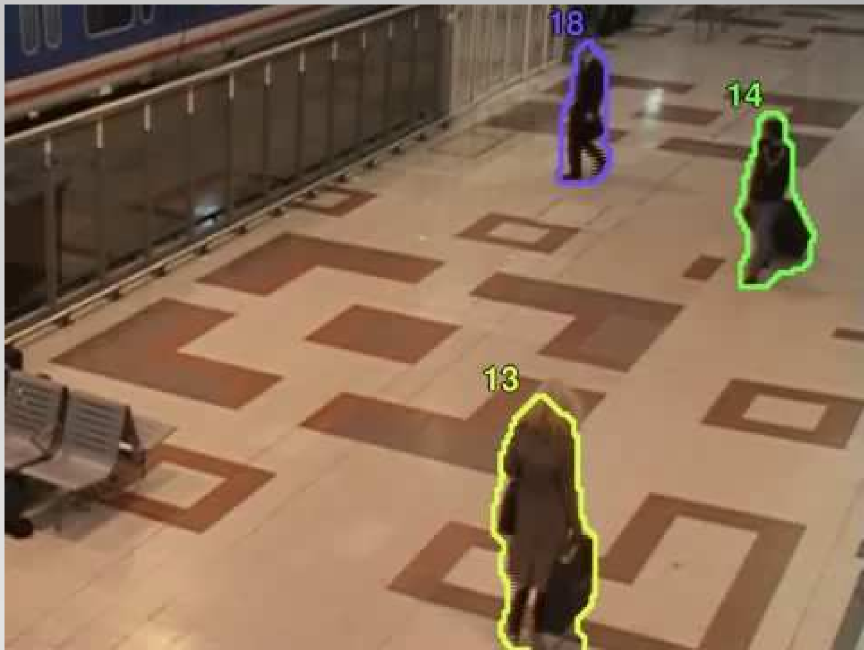
João F. Henriques   Rui Caseiro   Jorge Batista

Institute of Systems and Robotics
University of Coimbra

International Conference on Computer Vision, 2011

Motivation

- Tracking multiple people
- **Single, uncalibrated** camera
- Realistic video surveillance scenes



Problem far from solved!

- Tracking as global optimization
  1. Detect in all frames
  2. Optimize all tracks simultaneously (minimize a cost function)

Contributions:

Solid theoretical treatment of groups

$\Rightarrow$

Proposed method:
- Handles merged measurements
- Global optima in polynomial time

Previous work

Multiple-matching $\Rightarrow$ Target counts (group identification) $\Rightarrow$ Recovering identities

Previous work

	Multiple-matching ⇒	Target counts (group identification) ⇒	Recovering identities
C. Huang et al. ECCV'08	Global solution in $\mathcal{O}(n^3)$ No merges/splits	No groups	Obtained trivially
A. Perera et al. CVPR'06	Local solution in $\mathcal{O}(n^3)$ (Global in $\mathcal{O}(2^n)$)	Limited to groups of 2	Limited to groups of 2
J. Sullivan et al. ECCV/CVPR'06	Local solution in $\mathcal{O}(n)$	No convergence guarantees	Global solution in $\mathcal{O}(2^n)$
J. F. Henriques et al., ICCV'11	Global solution in $\mathcal{O}(n^3)$	Global solution in $\mathcal{O}(n^3)$	Global solution in $\mathcal{O}(n^3)$

Outline

1. Global optimization tracking review

2. Proposed method:

2.1. Multiple-matching ⇒ 2.2. Target counts (group identification) ⇒ 2.3. Recovering identities

3. Conclusion and results

Outline

1. Global optimization tracking review

2. Proposed method:

2.1. Multiple-matching ⇒ 2.2. Target counts (group identification) ⇒ 2.3. Recovering identities

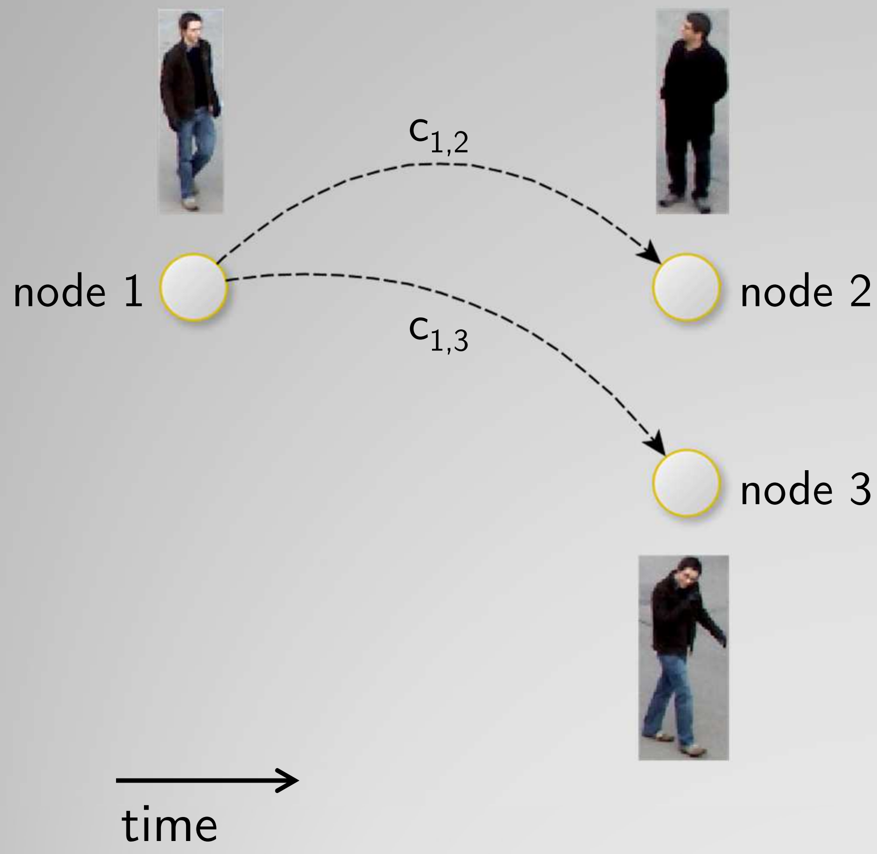
3. Conclusion and results

Global optimization tracking review

1. Detect and track with a simple method
   - Stop trackers when confidence is low
   - Output: **tracklets**
     (short track segments)

2. Each tracklet is a **node** in a graph

Global optimization tracking review



3. Connect each node to another if it occurs later in time

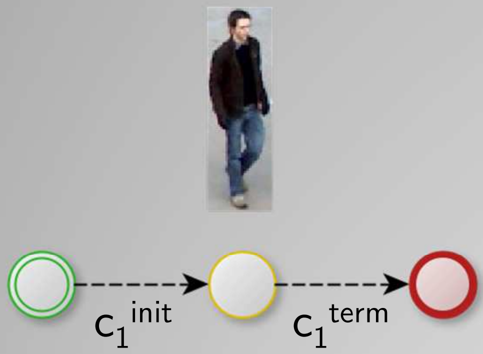
4. Calculate a **cost** for each of these **arcs**

   Low cost $\Rightarrow$ same target (hopefully!)

   Use distance, appearance similarity, size difference...

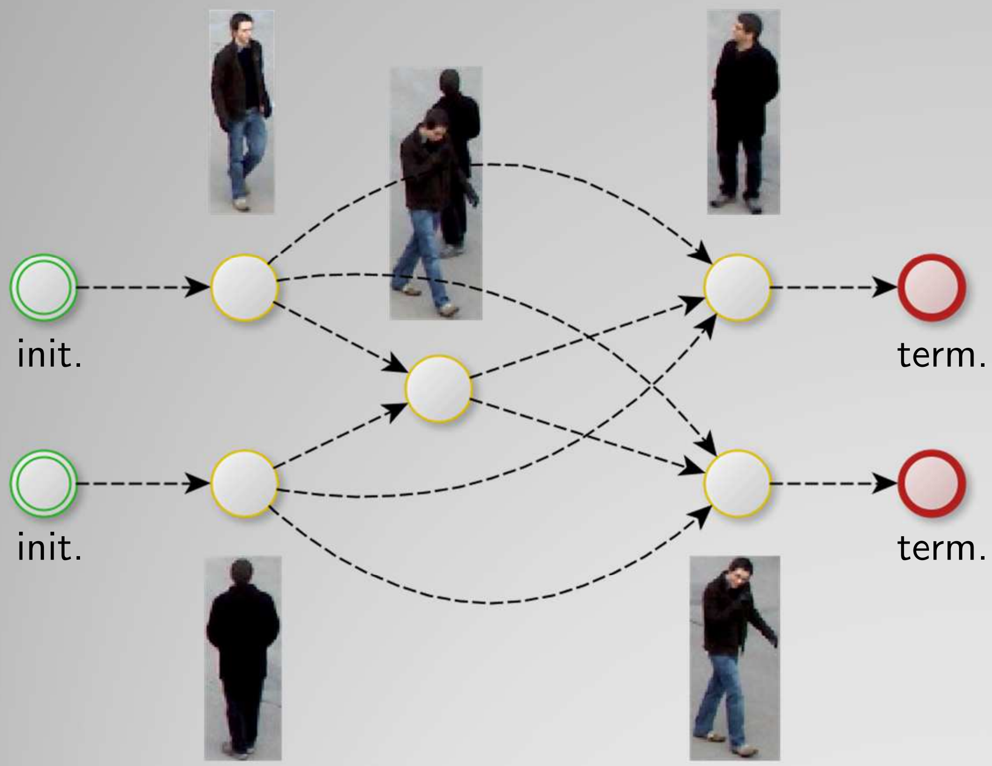
- These are **just hypotheses**.
- Optimization will choose the best set of arcs.

Global optimization tracking review



5. Termination nodes – hypothesis of ending a track
6. Initialization nodes – hypothesis of starting a track
7. Costs based on entry/exit locations model

Global optimization tracking review



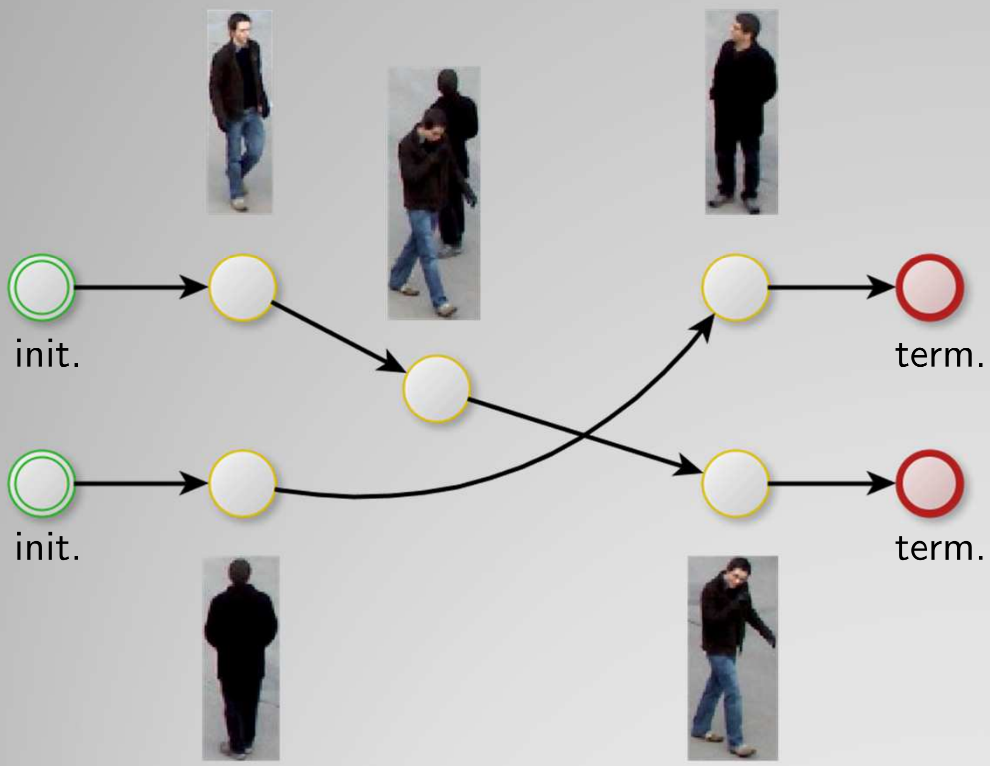
(Some init./term. nodes omitted for clarity)

- A typical graph:
  - Possibly occluded targets – long-range arcs
  - Optimization is free to choose between many hypotheses

Note:

- Long occlusions ⇒ many arcs ⇒ mistakes more likely (merged measurements solves this)

Global optimization tracking review



Optimal matching

- Find set of arcs X with a minimum total cost

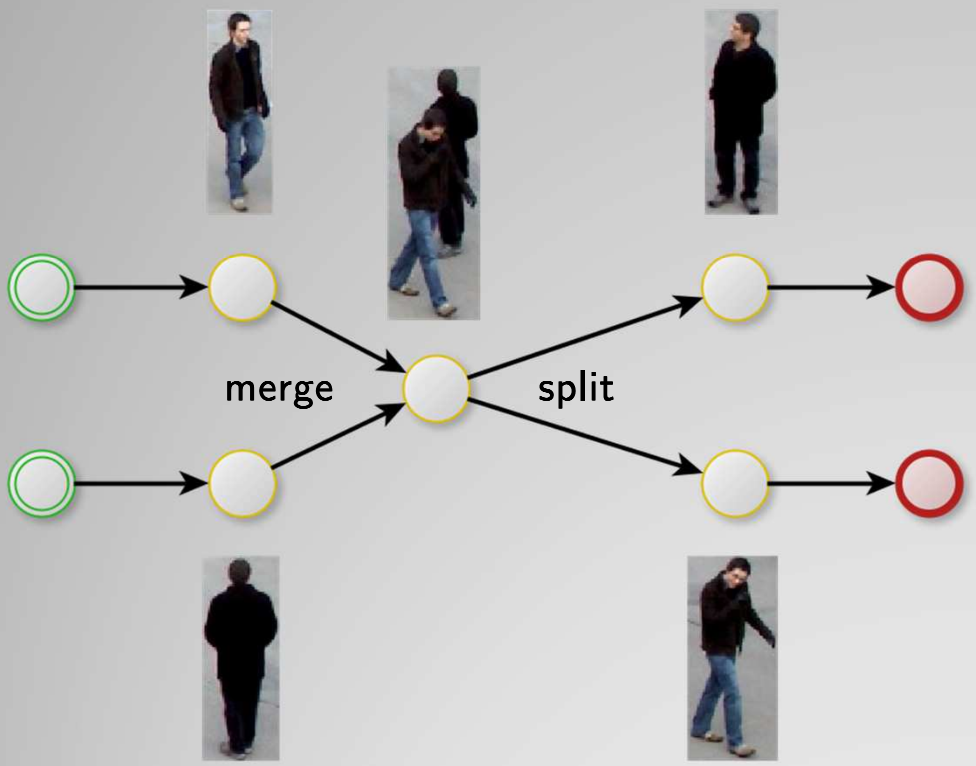
$$X^* = \arg\min_X \sum_{i,j} c_{ij} x_{ij}$$

- Subject to **no-overlap restriction**: only 1 ingoing and 1 outgoing arc in each node

$$s.t. \begin{cases} \sum_j x_{ij} = 1, \forall i \\ \sum_i x_{ij} = 1, \forall j \end{cases}$$

Hungarian Algorithm ⇒ Solution in $\mathcal{O}(n^3)$ (n: #nodes)

Merged measurements



- A **single node** may actually represent **several targets**.
- Merges and splits **violate** the no-overlap restriction.

⇓

Can't use Hungarian Algorithm directly!

(Cannot be solved with:
- Min. cost flow
- QP
- QBP)

Somehow...

Change graph to support:

All the previous arcs
+
Merges/splits hypotheses

Outline

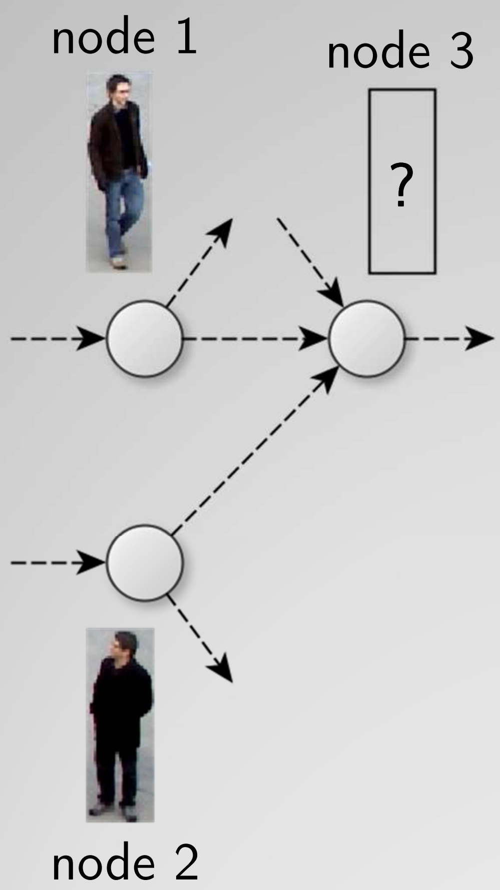
1. Global optimization tracking review

2. Proposed method:

2.1. Multiple-matching ⇒ 2.2. Target counts (group identification) ⇒ 2.3. Recovering identities

3. Conclusion and results

Multiple-matching



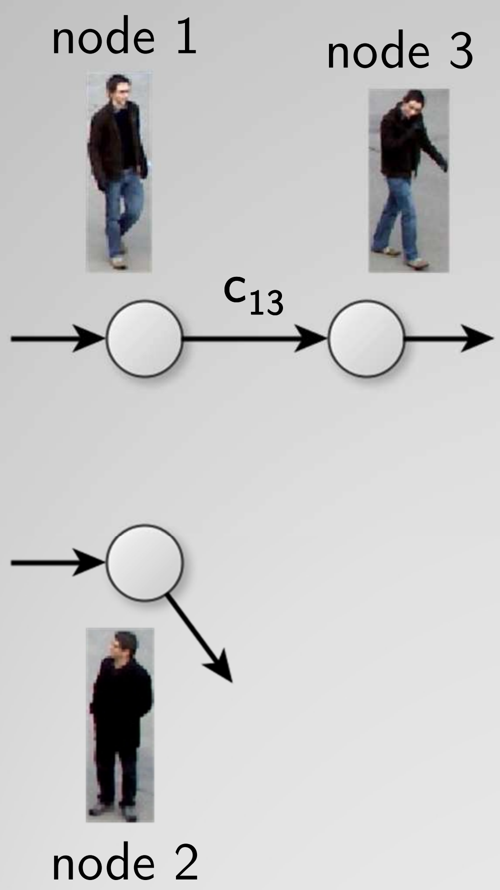
Don't know a-priori if node 3 is:

- Same target as node 1
- Same target as node 2
- Both targets (merged)

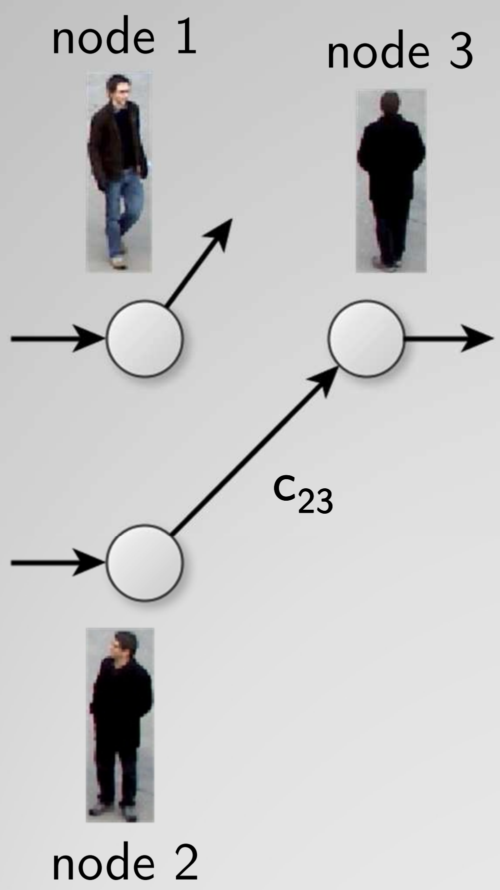
Can calculate a cost for each hypothesis:

$$c_{13}, \quad c_{23}, \quad c_{merge}$$

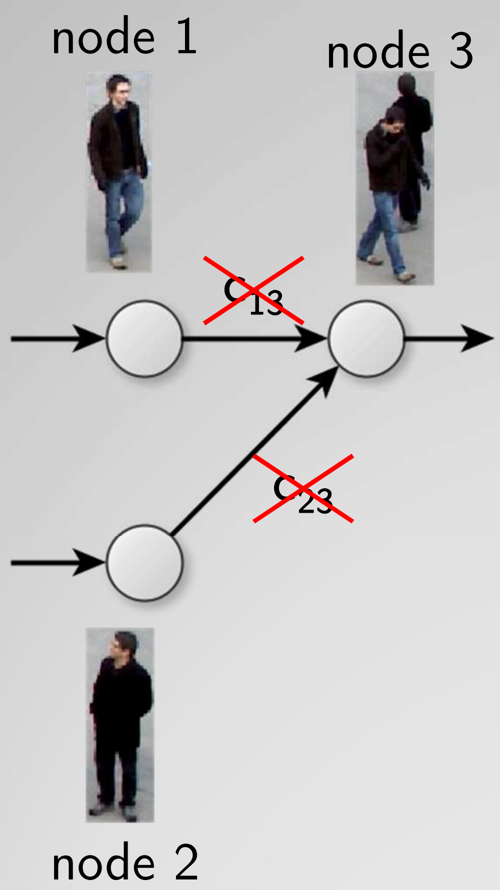
Multiple-matching



Multiple-matching



Multiple-matching



Total cost

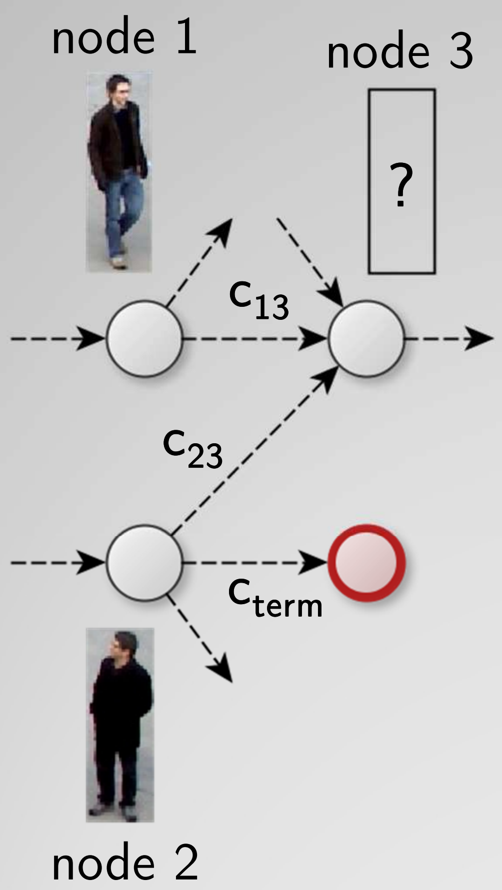
$= c_{merge}$

$\neq c_{13} + c_{23}$

Problems:

- Total cost is **not linear**
- **Violates** the no-overlap restriction

Multiple-matching



Key idea:

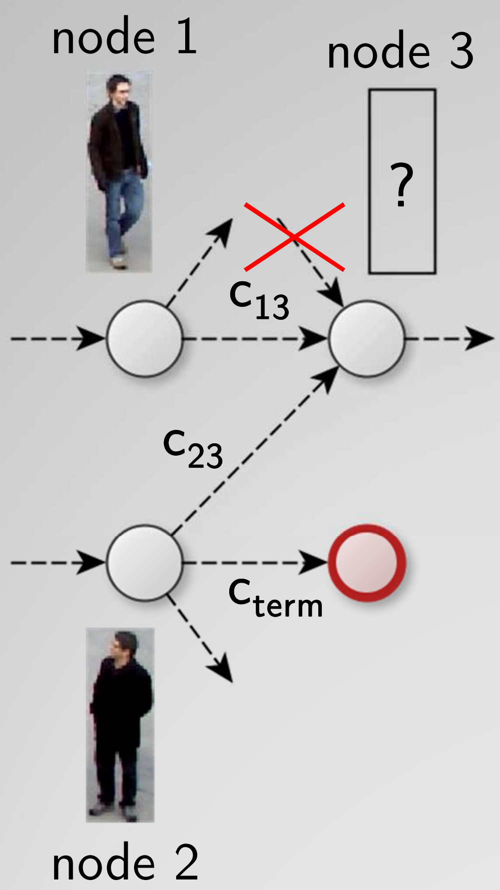
Encode a merge as one normal link and one termination.

Conditions:

- Total cost of merge must be c_{merge}

$\Rightarrow c_{term} = c_{merge} - c_{13}$

Multiple-matching



Key idea:

Encode a merge as one normal link and one termination.

Conditions:

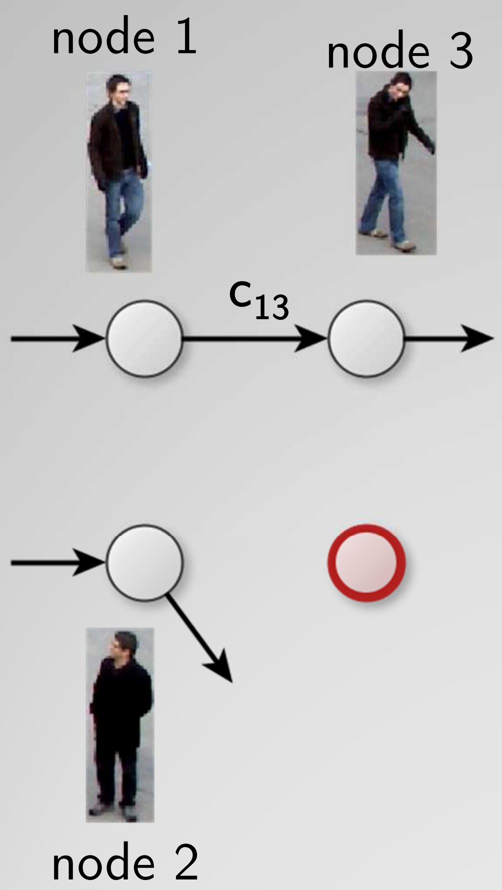
- Total cost of merge must be c_{merge}

  $\Rightarrow c_{term} = c_{merge} - c_{13}$

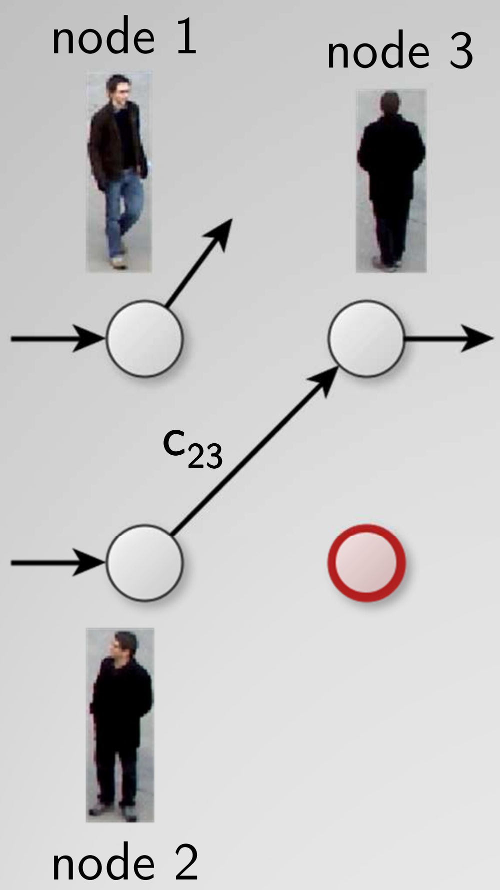
- If c_{term} is active, c_{13} must be active

  $\Rightarrow$ Disable all other incoming arcs to node 3

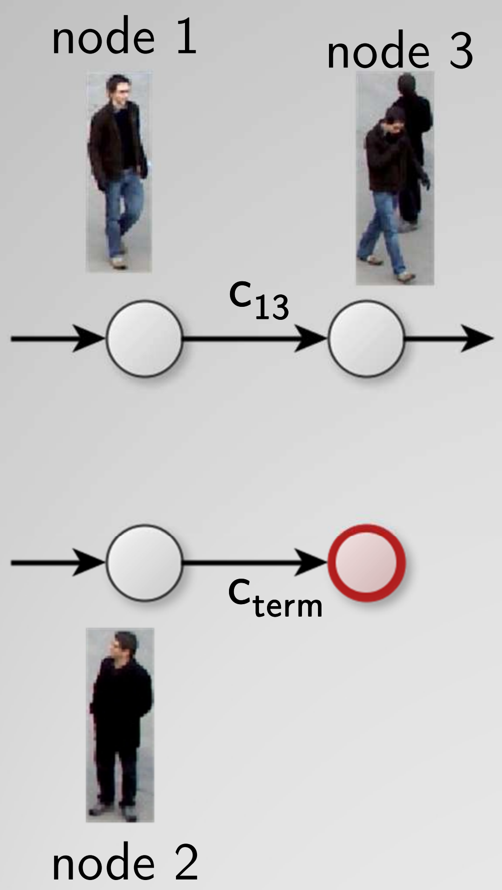
Multiple-matching



Multiple-matching



Multiple-matching



- Total cost

  $$= c_{13} + c_{term} = c_{merge}$$

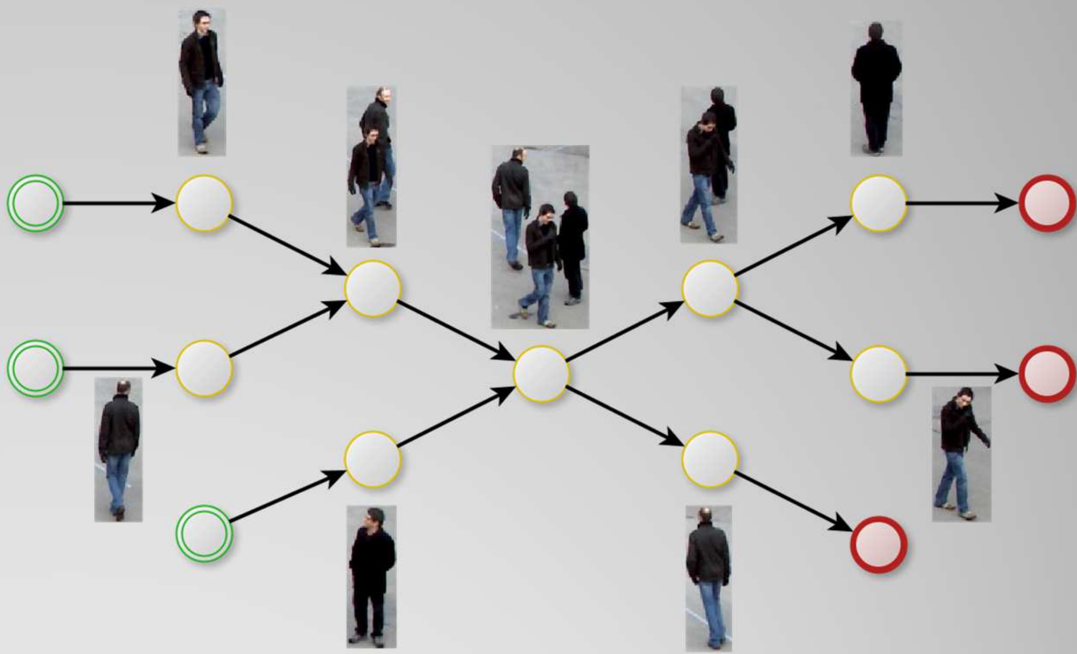
  as required.

- The proposed structure restricts the solutions to **only** these 3.

Multiple-matching

- For splits, reverse arc directions.
- The Hungarian Algorithm is free to choose
  - merges,
  - splits,
  - regular arcs

  that better explain data.
- Same computational complexity as simple matching.



Output

Outline

1. Global optimization tracking review

2. Proposed method:

2.1. Multiple-matching ⇒ 2.2. Target counts (group identification) ⇒ 2.3. Recovering identities

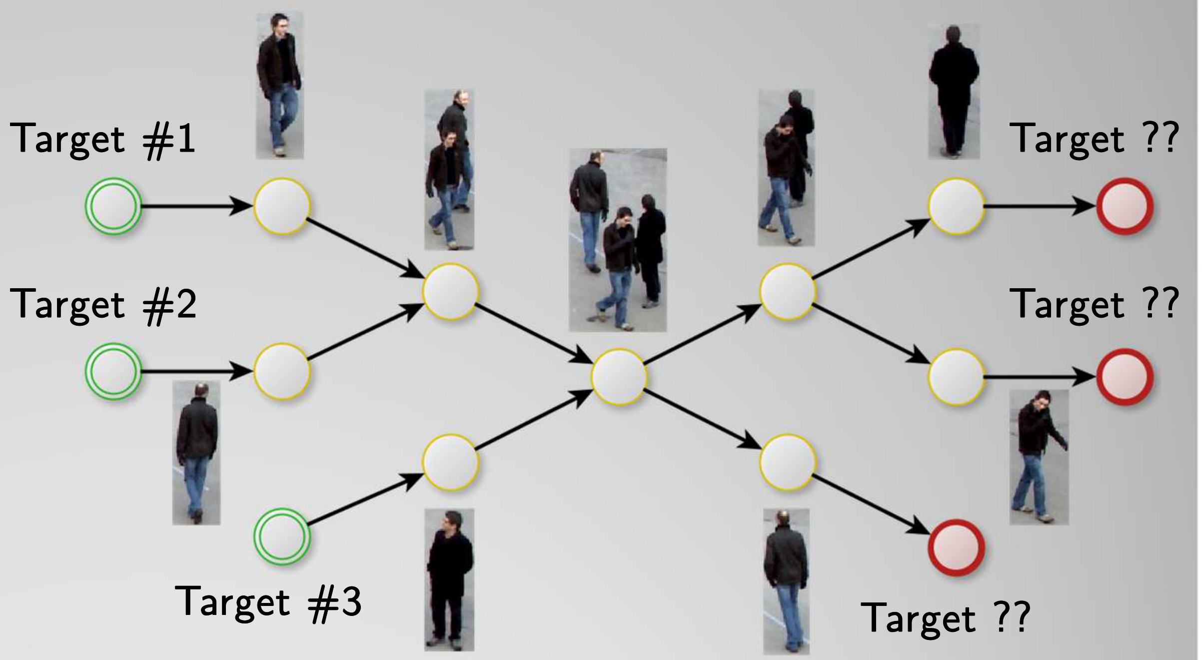
3. Conclusion and results

Groups

- Targets travelling alone can be tracked unambiguously.

But:

- Individual identities are lost in groups.



Strategy:

1. Identify nodes that correspond to groups.
2. Match individual targets across the groups.

Target counts as flow circulation

- To robustly identify groups, **count targets** in each node.
- Solve min-cost circulation:

$$f^* = \arg\min_{f} \sum_{j \in \mathcal{J}} \hat{c}_j f_j$$

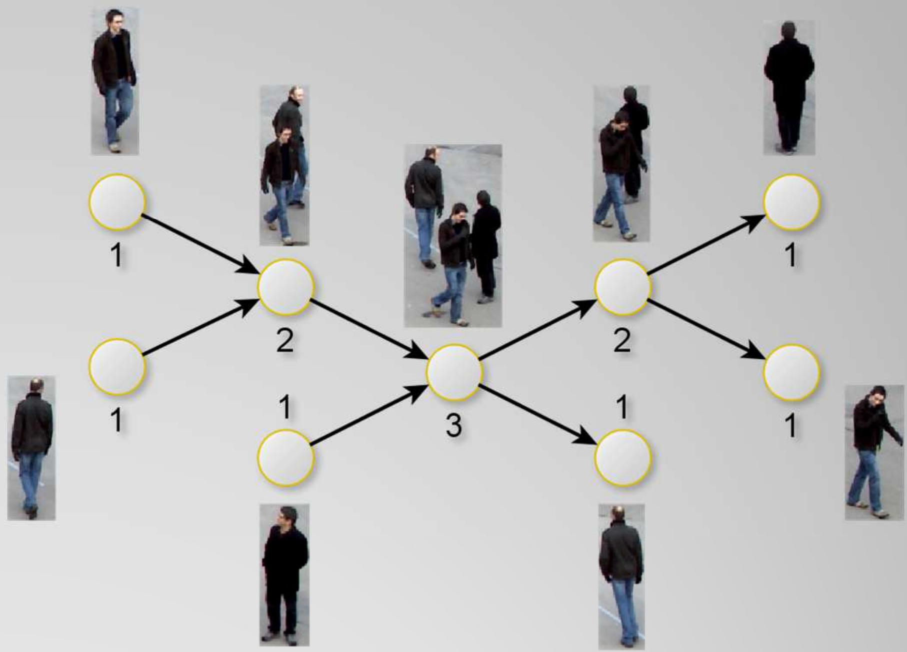
$$\text{s.t.:} \sum_{j \in out(i)} f_j - \sum_{j \in in(i)} f_j = 0, \ \forall i \in \mathcal{I}$$

$$1 \leq f_j \ , \forall j \in \mathcal{J}$$

Minimize number of targets

Flow conservation

Boundary conditions (at least 1 target per node)



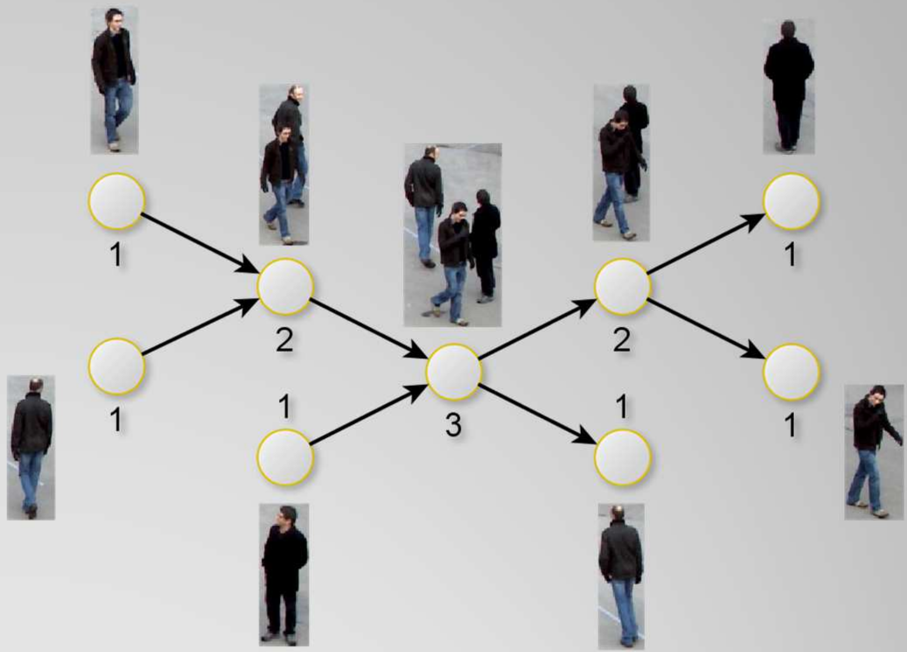
Solved in polynomial time with the Edmonds-Karp algorithm.

Target counts as flow circulation

- Costs inversely proportional to box size
  (larger boxes ⇒ larger groups)
  ↳ Mostly as a tie-breaker

- Counts obtained **solely from the graph topology.**

- Equivalent to locating occluded targets.



Outline

1. Global optimization tracking review

2. Proposed method:

2.1. Multiple-matching ⇒ 2.2. Target counts (group identification) ⇒ 2.3. Recovering identities

3. Conclusion and results

Recovering identities

Groups subgraph $\mathcal{G}_{group}$

$=\{$arcs and nodes with flow $> 1\}$

$$\mathcal{T}_{group} = \left\{ T_i \mid \sum_{j \in in(i)} f_j^* > 1,\, T_i \in \mathcal{T} \right\}$$

Entering a group

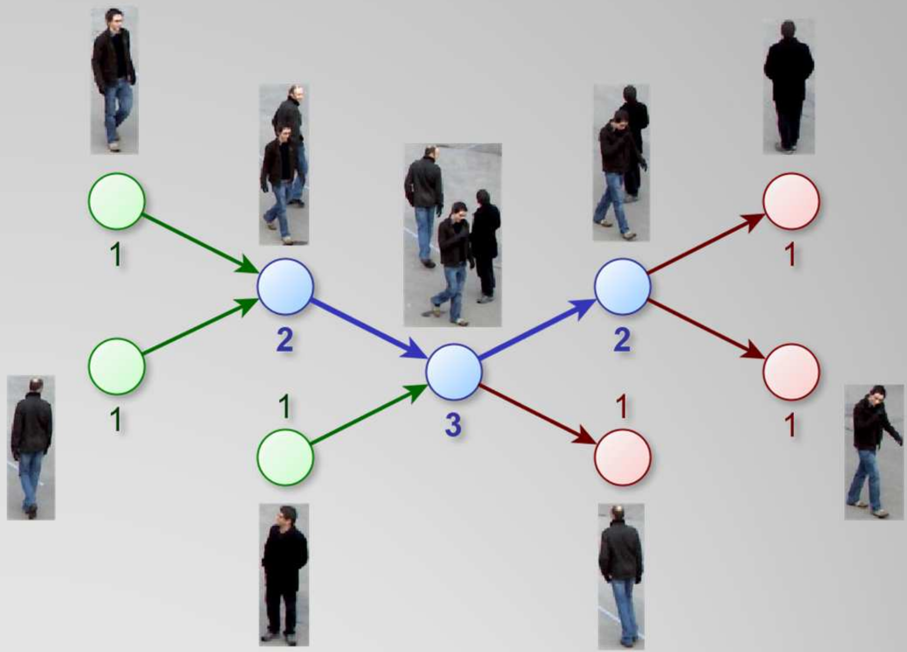
$=\{$arcs pointing to $\mathcal{G}_{group}\}$

$$\mathcal{T}_{in} = \{T_i \mid \exists (T_i, T_k) \in \mathcal{A},\, T_i \notin \mathcal{T}_{group},\, T_k \in \mathcal{T}_{group}\}$$

Exiting a group

$=\{$arcs pointing from $\mathcal{G}_{group}\}$

$$\mathcal{T}_{out} = \{T_i \mid \exists (T_k, T_i) \in \mathcal{A},\, T_i \notin \mathcal{T}_{group},\, T_k \in \mathcal{T}_{group}\}$$

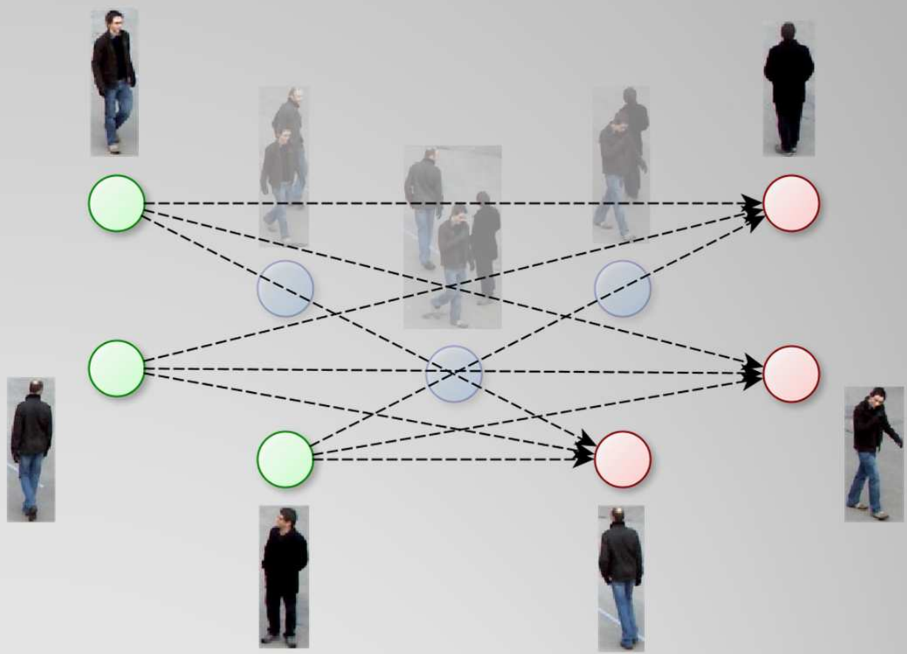


To track across groups, match targets entering to targets exiting. $\Rightarrow$ Simple matching problem

Recovering identities

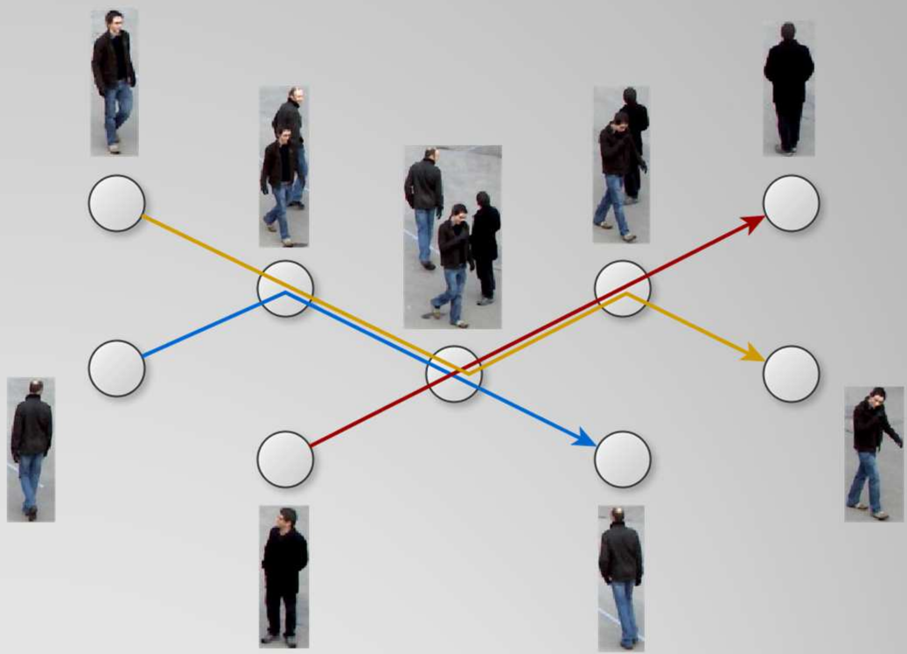
- Match across groups:

  1. Connect a node in entering to a node in exiting if it's reachable through $\mathcal{G}_{group}$.
  2. Calculate costs as usual.
  3. Optimal matching with Hungarian Algorithm.



Recovering identities

- Unlike other methods, we **don't link across arbitrary gaps**.
- Target's trajectory is **restricted** to group's trajectory.
- Only admits **physically plausible** tracks for occlusions.



Outline

1. Global optimization tracking review

2. Proposed method:

2.1. Multiple-matching ⇒ 2.2. Target counts (group identification) ⇒ 2.3. Recovering identities

3. Conclusion and results

Results

PETS 2009 (single view)
Outperforms **all 12** state-of-the-art systems at PETS 2010.

PETS 2006 (single view)
Good results, challenging videos.
No other MOTA scores to compare.

Detection method:
foreground segmentation

- Simple to implement
- **Frequent merges and splits**
  ⇒ Stress-test the framework

Metric	PETS'09	PETS'06		
	S2-L1	S1-T1	S3-T7	S4-T5
MOTA	0.966	0.785	0.816	0.883
Mismatches	10	16	4	7
Precision	0.985	0.882	0.847	0.932
Recall	0.986	0.908	0.997	0.954

Results

Conclusion

Formalized group formation
as a global optimization problem.

Polynomial-time algorithms for:

- Multi-match problem.
- Counting problem.
- Identity match across groups.

Improves state-of-the-art
with simple detection and low-level stages.

Questions?