A Flexible Tensor Block Coordinate Ascent Scheme for Hypergraph Matching

max planck institut informatik — Universität des Saarlandes — CVPR 2015 Boston, June 7-12

Quynh Nguyen, Antoine Gautier and Matthias Hein

Max Planck Institute for Informatics, Saarland University, Germany

Graph Matching in Computer Vision

Applications

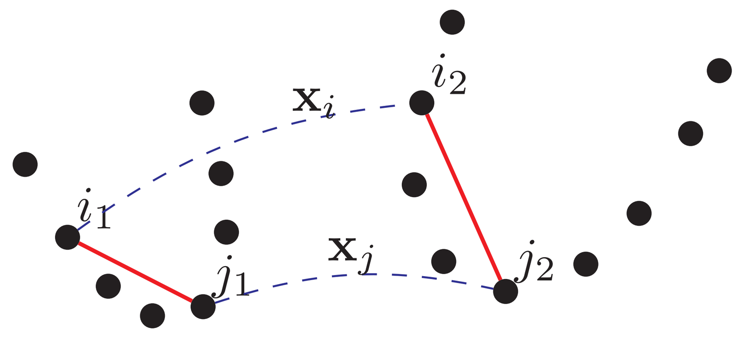
1. Object Detection
2. Action Recognition
3. Feature Correspondence, etc

Why is this difficult?

1. **outliers** and **deformation** in data
2. **geometric transformations** involved
3. the optimization problem is **NP-Hard**

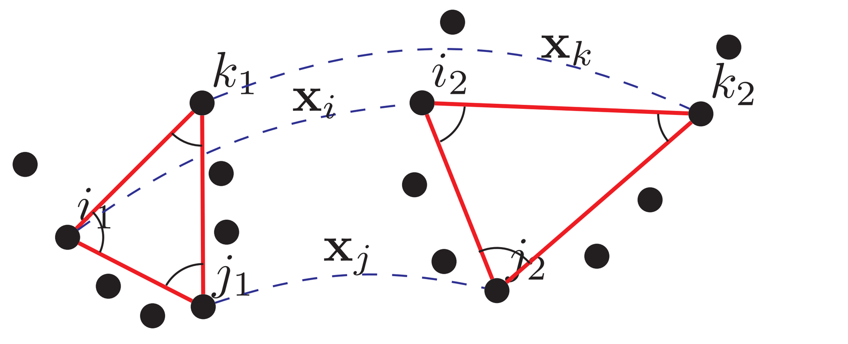
Why Hypergraphs ?

The set of assignment matrices: $M := \{\mathbf{X} \in \{0,1\}^{n_1 \times n_2} \mid \mathbf{X}\mathbf{1}_{n_2} = \mathbf{1}_{n_1}, \mathbf{X}^T\mathbf{1}_{n_1} \le \mathbf{1}_{n_2}\}$.



$$\max_{\mathbf{x}\in M} \sum_{ij} \mathcal{F}_{ij}\mathbf{x}_i\mathbf{x}_j$$

- limited to **pairwise relations**
- difficult to deal with rotation + scale



$$\max_{\mathbf{x}\in M} F^3(\mathbf{x},\mathbf{x},\mathbf{x}) = \max_{\mathbf{x}\in M} \sum_{i,j,k} \mathcal{F}^3_{ijk}\mathbf{x}_i\mathbf{x}_j\mathbf{x}_k$$

- consider **higher order relations**
- scale and rotation invariant, etc

Previous Work

$\max_{\mathbf{x}\in M} F^3(\mathbf{x},\mathbf{x},\mathbf{x})$

Tensor Marginalization
Zass and Shashua, CVPR' 08
Chertok and Keller, PAMI' 10
(loss of information)

Spectral Relaxation
Duchenne *et al.*, CVPR' 09
(loose)

Continuous Relaxation
$\mathbf{x} \in [0,1]^{n_1\times n_2}, \mathbf{x}\mathbf{1}_{n_2} = \mathbf{1}_{n_1}, \mathbf{x}^T\mathbf{1}_{n_1} \le \mathbf{1}_{n_2}$
Lee *et al.*, CVPR' 11

→ Continuous Solution ($\mathbf{x} \in [0,1]^{n_1\times n_2}$)

→ Discretization
(potential loss of accuracy)

→ Discrete Solution ($\mathbf{x} \in M$)

Our Contribution

- A flexible & theoretically sound **optimization framework** for hypergraph matching
- **Two new algorithms** which are robust to noise & transformations
- **Monotonic ascent** for original score function on the set of assignment matrices M
- **Convergence** guaranteed in finite number of iterations
- **Outperform** previous hypergraph matching as well as graph matching algorithms

Main Ideas: Multilinear Reformulation ⟶ Block Coordinate Ascent

Proposed Framework

Associate to an m-order tensor $\mathcal{F}^m$ a multilinear form $F^m : \mathbb{R}^n \times \ldots \times \mathbb{R}^n \to \mathbb{R}$ such that $F^m(\mathbf{x}^1,\ldots,\mathbf{x}^m) = \sum_{i_1,\ldots,i_m} \mathcal{F}^m_{i_1,\ldots,i_m}\mathbf{x}^1_{i_1}\ldots\mathbf{x}^m_{i_m}$.

1. **Lifting the tensor to 4th order tensor:** $\mathcal{F}^4_{ijkl} = \mathcal{F}^3_{ijk} + \mathcal{F}^3_{ijl} + \mathcal{F}^3_{ikl} + \mathcal{F}^3_{jkl}$
$$F^4(\mathbf{x},\mathbf{x},\mathbf{x},\mathbf{x}) = 4F^3(\mathbf{x},\mathbf{x},\mathbf{x})\sum_i \mathbf{x}_i \implies \max_{\mathbf{x}\in M} F^4(\mathbf{x},\mathbf{x},\mathbf{x},\mathbf{x}) \equiv \max_{\mathbf{x}\in M} F^3(\mathbf{x},\mathbf{x},\mathbf{x})$$
2. **Convexifying the multilinear form**
$$F^4_\alpha(\mathbf{x},\mathbf{y},\mathbf{z},\mathbf{t}) := F^4(\mathbf{x},\mathbf{y},\mathbf{z},\mathbf{t}) + \alpha\frac{\langle\mathbf{x},\mathbf{y}\rangle\langle\mathbf{z},\mathbf{t}\rangle + \langle\mathbf{x},\mathbf{z}\rangle\langle\mathbf{y},\mathbf{t}\rangle + \langle\mathbf{x},\mathbf{t}\rangle\langle\mathbf{y},\mathbf{z}\rangle}{3}$$
$$\max_{\mathbf{x}\in M} F^4_\alpha(\mathbf{x},\mathbf{x},\mathbf{x},\mathbf{x}) \equiv \max_{\mathbf{x}\in M} F^4(\mathbf{x},\mathbf{x},\mathbf{x},\mathbf{x})$$
3. **Multilinear Optimization**
For any $\alpha \ge 3\sqrt{\sum_{i,j,k,l=1}^n (\mathcal{F}^4_{ijkl})^2}$, $F^4_\alpha(\mathbf{x},\mathbf{x},\mathbf{x},\mathbf{x})$ will be convex and it holds:
   - Monotonic ascent for multilinear form implies the same for score function
$$F^4_\alpha(\mathbf{x},\mathbf{y},\mathbf{z},\mathbf{t}) \le \max_{\mathbf{u}\in\{\mathbf{x},\mathbf{y},\mathbf{z},\mathbf{t}\}} F^4_\alpha(\mathbf{u},\mathbf{u},\mathbf{u},\mathbf{u}), \quad \forall \mathbf{x},\mathbf{y},\mathbf{z},\mathbf{t} \in \mathbb{R}^n$$
$$F^4_\alpha(\mathbf{x},\mathbf{x},\mathbf{y},\mathbf{y}) \le \max_{\mathbf{u}\in\{\mathbf{x},\mathbf{y}\}} F^4_\alpha(\mathbf{u},\mathbf{u},\mathbf{u},\mathbf{u}), \quad \forall \mathbf{x},\mathbf{y} \in \mathbb{R}^n.$$
   - The following optimization problems are all equivalent
$$\max_{\mathbf{x}\in M} F^4_\alpha(\mathbf{x},\mathbf{x},\mathbf{x},\mathbf{x}) = \max_{\mathbf{x},\mathbf{y}\in M} F^4_\alpha(\mathbf{x},\mathbf{x},\mathbf{y},\mathbf{y}) = \max_{\mathbf{x},\mathbf{y},\mathbf{z},\mathbf{t}\in M} F^4_\alpha(\mathbf{x},\mathbf{y},\mathbf{z},\mathbf{t})$$

$\max_{\mathbf{x}\in M} F^3(\mathbf{x},\mathbf{x},\mathbf{x})$ → $\max_{\mathbf{x}\in M} F^4(\mathbf{x},\mathbf{x},\mathbf{x},\mathbf{x})$ → $\max_{\mathbf{x}\in M} F^4_\alpha(\mathbf{x},\mathbf{x},\mathbf{x},\mathbf{x})$ → $\max_{\mathbf{x},\mathbf{y},\mathbf{z},\mathbf{t}\in M} F^4_\alpha(\mathbf{x},\mathbf{y},\mathbf{z},\mathbf{t})$ → Algorithm 1
→ $\max_{\mathbf{x},\mathbf{y}\in M} F^4_\alpha(\mathbf{x},\mathbf{x},\mathbf{y},\mathbf{y})$ → Algorithm 2

Algorithm 1.
repeat
1 Update $(\mathbf{x},\mathbf{y},\mathbf{z},\mathbf{t})$ in cyclic order using BCA
  if *no further ascent* **then**
    $\mathbf{u} = \arg\max_{\mathbf{u}\in\{\mathbf{x},\mathbf{y},\mathbf{z},\mathbf{t}\}} F^4_\alpha(\mathbf{u},\mathbf{u},\mathbf{u},\mathbf{u})$
    $\mathbf{x} = \mathbf{y} = \mathbf{z} = \mathbf{t} = \mathbf{u}$
until *no possible ascent*

Algorithm 2.
repeat
1 Update $(\mathbf{x},\mathbf{y})$ in cyclic order using BCA
  if *no further ascent* **then**
    $\mathbf{u} = \arg\max_{\mathbf{u}\in\{\mathbf{x},\mathbf{y}\}} F^4_\alpha(\mathbf{u},\mathbf{u},\mathbf{u},\mathbf{u})$
    $\mathbf{x} = \mathbf{y} = \mathbf{u}$
until *no possible ascent*

Attributes

1. Monotonic ascent for $F^3(\mathbf{x},\mathbf{x},\mathbf{x})$ on M
2. Converge in finite number of iterations
3. No need for discretization at final step
4. Reuse of existing lower order methods
5. Adaptable to other constraint sets

Source Code

The MATLAB/C code can be downloaded at:
`http://www.ml.uni-saarland.de/people/nguyen.htm`

Matching Point Sets in $\mathbb{R}^2$

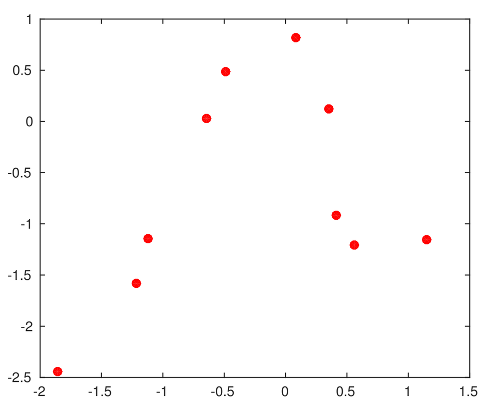
Graph Matching Methods

- SM: Spectral Matching
- IPFP: Integer Projected Fixed Point
- RRWM: Reweighted Random Walk
- MPM: Max Pooling Matching

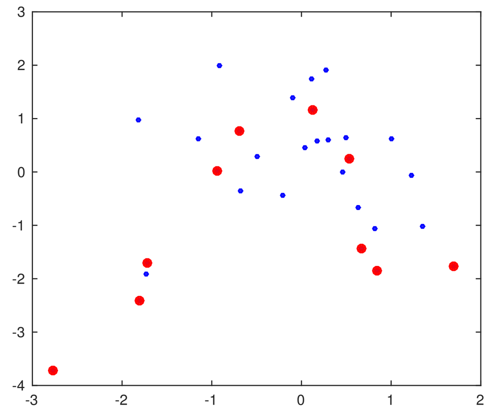
Hypergraph Matching Methods

- HGM: Marginalization Approach
- TM: Tensor Matching
- RRWHM: Extension of RRWM
- BCAGM(s): Our Algorithms

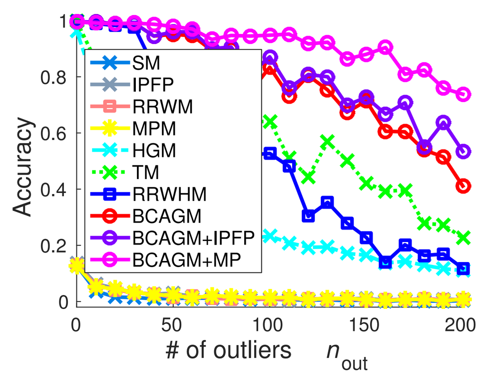
Experiment Setup: 10 inliers, additive noise $\mathcal{N}(0,\sigma^2)$ with $\sigma = 0.3$, $scale = 1.5$

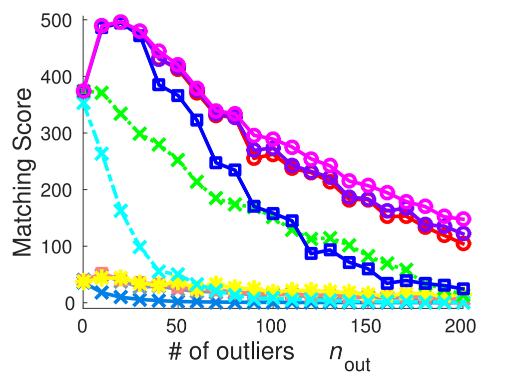


1st Point Set

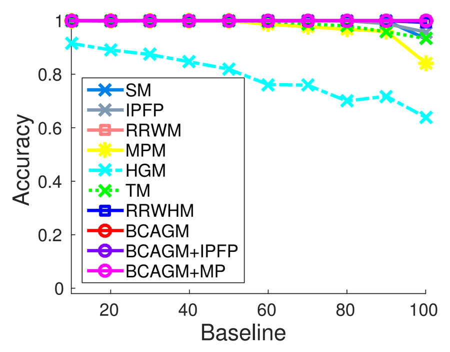


2nd Point Set (20 outliers)

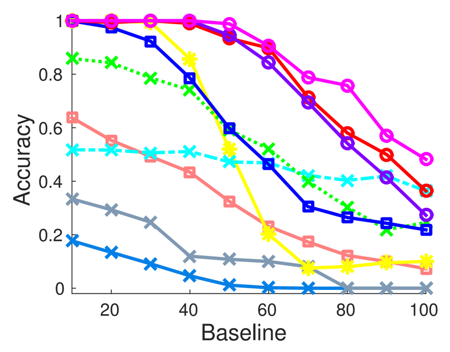




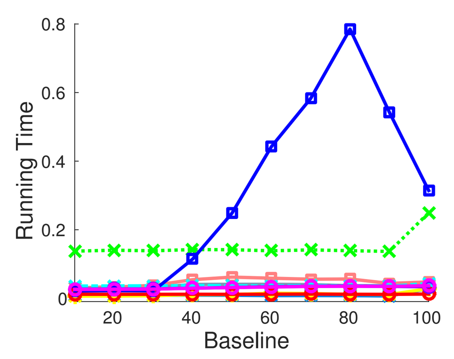
CMU House Dataset

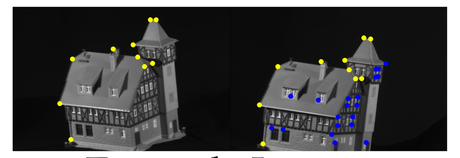


30 pts vs. 30 pts

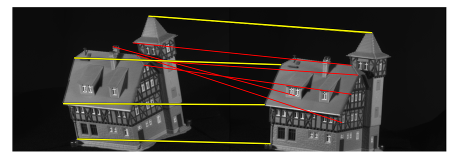


10 pts vs. 30 pts

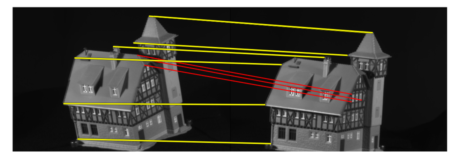




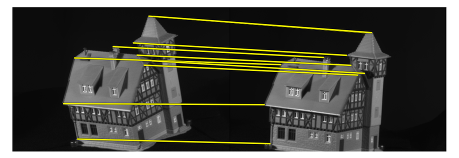
Example Input



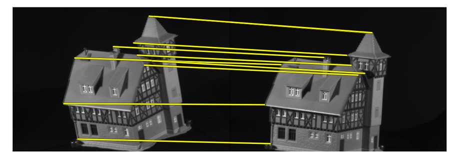
TM (5/10)



RRWHM (7/10)

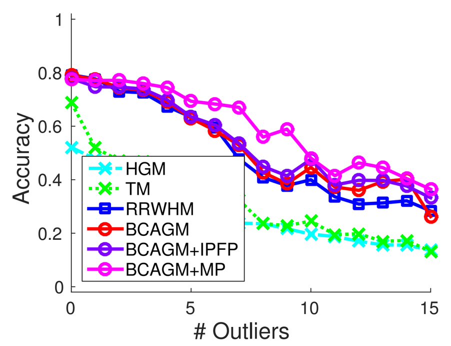


BCAGM (10/10)

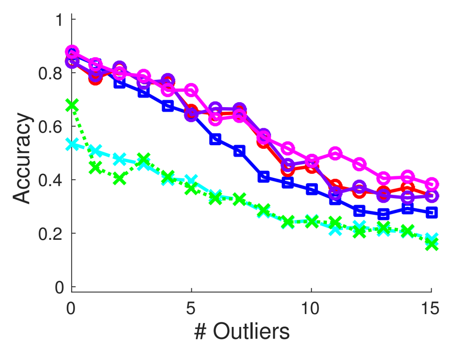


BCAGM+MP (10/10)

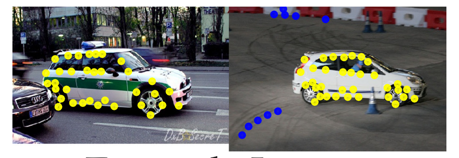
Car and Motorbike Dataset



Car

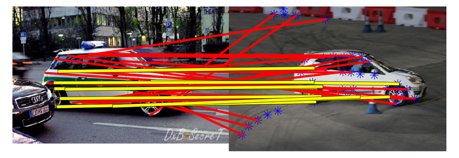


Motorbike

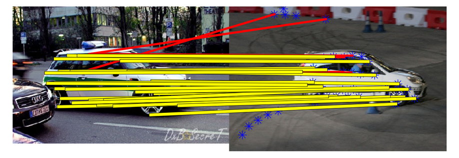


Example Input

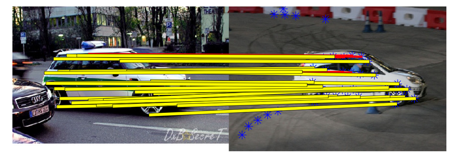
TM (10/34)



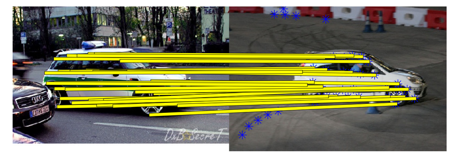
HGM (9/34)



RRWHM (28/34)



BCAGM (28/34)



BCAGM+MP (34/34)