Graphical Models for Structural Pattern Recognition

Tibério Caetano

National ICT Australia

(thanks to collaborators T. Caelli, D. Schuurmans and D. Barone)

Machine Learning Summer School, Canberra, February 2006
Outline

- Structural Pattern Recognition (Matching)
- Matching as Inference in Graphical Models
- Problem 1: Exact rigid point matching
- Problem 2: Attributed graph matching
- Problem 3: Invariant point matching
Structural Pattern Recognition
Structural Pattern Recognition
| a | 1 |
| b | 2 |
| c | 3 |
| d | 4 |
| e | 5 |
| f | 6 |
| g | 7 |
| h | 8 |
| i | 9 |
| j | 10 |
| k | 11 |
| l | 12 |
| m | 13 |
| n | 14 |
| o | 15 |
| p | 16 |
| q | 17 |
| r | 18 |
| s | 19 |
| t | 20 |
| u | 21 |
| v | 22 |
| w | 23 |
| x | 24 |
| y | 25 |
| z | 26 |
| aa | 27 |
| ab | 28 |
| ac | 29 |
Structural Pattern Recognition

$S^T$ possible maps

$S$ - Scene size \hspace{1cm} $T$ - Template size
Costs:

\[ U(X = x) = \sum_i U_i(X_i; x_{f(i)}) \]

tractable

\[ U(X = x) = \sum_i U_i(X_i; x_{f(i)}) + \sum_i \sum_j U_{ij}(X_i, X_j; x_{f(i)}, x_{f(j)}) \]

NP-hard

\[ U(X = x) = \sum_{n=1}^{N} \sum_{i_1} \cdots \sum_{i_n} U_{i_1i_2\cdots i_n}(X_{i_1}, X_{i_2}, \ldots, X_{i_n}; x_{f(i_1)}, x_{f(i_2)}, \ldots, x_{f(i_N)}) \]

NP-hard
Structural Pattern Recognition

Typical solution:
assume quadratic assignment and optimize using heuristics - suboptimal

Examples of algorithms: relaxation labeling, graduated assignment
Matching as Inference in Graphical Models
Proposed solution:

- Impose **structured approximation** on the cost function according to a probabilistic graphical model

- Solve the assignment problem **exactly** in the resulting tractable model

- Show that the resulting tractable model **preserves** the solution of the complete model (which is not tractable)
Matching as Inference in Graphical Models

It is possible to represent the optimizer of the cost

$$U(X = x) = \sum_{n=1}^{N} \sum_{i_1} \cdots \sum_{i_n} U_{i_1i_2\ldots i_n}(X_{i_1}, X_{i_2}, \ldots, X_{i_n}; x_{f(i_1)}, x_{f(i_2)}, \ldots, x_{f(i_N)})$$

As the MAP assignment of a fully connected Markov random field:
Matching as Inference in Graphical Models

Note that this is also the case even for pairwise costs:

\[ U(X = x) = \sum_i U_i(X_i; x_{f(i)}) + \sum_i \sum_j U_{ij}(X_i, X_j; x_{f(i)}, x_{f(j)}) \]
Introduce structured MRF

\[ U(X = x) = \sum_i U_i(X_i; x_f(i)) + \sum_i \sum_j U_{ij}(X_i, X_j; x_f(i), x_f(j)) \]
We will see that:

1. The resulting models are exactly solvable in polynomial time via the Junction Tree algorithm

2. In the absence of noise, the optimal solutions in the resulting models are the same as those in the fully connected models for the problems studied

Result: optimal solutions in polynomial time
One-slide tutorial on the Junction Tree algorithm:

- The Junction Tree Algorithm is the algorithm used for exact inference in arbitrary Graphical Models

- It can only be applied after triangulating the graph

- Overall complexity is $TS^{k-1}$, where $T$ is the number of nodes, $S$ the number of realizations and $k - 1$ the size of the maximal clique of the graph after triangulation
Matching as Inference in Graphical Models

A -- B
C -- D
E -- F

A -- B
C -- D
E

ABC
BCD
CDE
DEF

BC
CD
DE
Problem 1: Rigid Point Matching
Problem 1: Rigid Point Matching

$$EDM_1 = \begin{bmatrix} \text{dist}^{11} & \text{dist}^{12} & \ldots & \text{dist}^{1T} \\ \text{dist}^{21} & \text{dist}^{22} & \ldots & \text{dist}^{2T} \\ \vdots & \vdots & \ddots & \vdots \\ \text{dist}^{T1} & \text{dist}^{T2} & \ldots & \text{dist}^{TT} \end{bmatrix}$$

$$EDM_2 = \begin{bmatrix} \text{dist}^{f(1)f(1)} & \text{dist}^{f(1)f(2)} & \ldots & \text{dist}^{f(1)f(T)} \\ \text{dist}^{f(2)f(1)} & \text{dist}^{f(2)f(2)} & \ldots & \text{dist}^{f(2)f(T)} \\ \vdots & \vdots & \ddots & \vdots \\ \text{dist}^{f(T)f(1)} & \text{dist}^{f(T)f(2)} & \ldots & \text{dist}^{f(T)f(T)} \end{bmatrix}$$
Problem 1: Rigid Point Matching

Problem Definition:

Given two point sets \( \mathbb{T} = \{d_i, i = 1, 2, \ldots, T\} \) and \( \mathbb{S} = \{c_j, j = 1, 2, \ldots, S\} \) in \( \mathbb{R}^n \) \((n \in \mathbb{N})\), find the function \( f : \mathbb{T} \rightarrow \mathbb{S} \) that minimizes

\[
U_T(f) = \sum_{i=1}^{T} \sum_{j=1}^{T} D(\text{dist}_1^{ij}, \text{dist}_2^{f(i)f(j)}),
\]

where \( D(d_{ij}, d_{f(i)f(j)}) \) is some dissimilarity measure between the distances \( \text{dist}_1^{ij} := \|d_i - d_j\| \) and \( \text{dist}_2^{f(i)f(j)} := \|c_{f(i)} - c_{f(j)}\| \)
Problem 1: Rigid Point Matching

Which structured approximation to use?
Problem 1: Rigid Point Matching

Some Geometry
Problem 1: Rigid Point Matching

Some Geometry
Problem 1: Rigid Point Matching

Some Geometry
Problem 1: Rigid Point Matching

Some Geometry
Problem 1: Rigid Point Matching

Some Geometry
Problem 1: Rigid Point Matching

Some Geometry
Problem 1: Rigid Point Matching

Some Geometry
Problem 1: Rigid Point Matching

Some Geometry
Problem 1: Rigid Point Matching

Some Geometry
Problem 1: Rigid Point Matching
Problem 1: Rigid Point Matching

- The resulting graph is **globally rigid**
- The resulting graph is technically a **3-tree**
- A 3-tree has a maximal clique size **fixed in 4**
- A 3-tree is **already** triangulated
Problem 1: Rigid Point Matching

This can be generalized to any dimensionality

In general:

- The graph is a $k$-tree in the case of $\mathbb{R}^{k-1}$
- The maximal clique size is fixed in $k + 1$
Problem 1: Rigid Point Matching

- Junction Tree algorithm minimizes:

\[ U_{G_d^{kt}}(f) = \sum_{i,j|d_{ij} \in E_d^{kt}} D(||d_i - d_j||, ||c_{f(d_i)} - c_{f(d_j)}||), \text{ for a } k\text{-tree graphical model} \]

\[ U_T(f) = \sum_{i=1}^{T} \sum_{j=1}^{T} D(||d_i - d_j||, ||c_{f(d_i)} - c_{f(d_j)}||), \text{ for a fully connected graphical model} \]
Problem 1: Rigid Point Matching

- **Theorem:** In the exact matching case, a mapping \( f \) which minimizes \( U_{G_{d}^{k}}(f) \) also minimizes \( U_{T}(f) \)
Problem 1: Rigid Point Matching

Diagram:

- $X_1$ connected to $X_2$, $X_3$, $X_4$, and $X_5$
- $X_2$ connected to $X_3$, $X_4$, and $X_5$
- $X_3$ connected to $X_4$ and $X_5$
- $X_4$ connected to $X_1$, $X_2$, and $X_3$
- $X_5$ connected to $X_1$, $X_2$, and $X_3$

Dotted lines indicate additional connections not explicitly shown in the diagram.
Problem 1: Rigid Point Matching

Propagation in the Junction Tree:

\[ X_1 \ X_2 \ X_3 \ X_4 \quad X_1 \ X_2 \ X_3 \ X_5 \quad X_1 \ X_2 \ X_3 \ X_{T-1} \quad X_1 \ X_2 \ X_3 \ X_T \]
Problem 1: Rigid Point Matching

Propagation in the Junction Tree:

\[
X_1 X_2 X_3 X_4 \quad X_1 X_2 X_3 X_5 \quad X_1 X_2 X_3 X_{T-1} \quad X_1 X_2 X_3 X_T
\]

\[
X_1 X_2 X_3 \quad \cdots \quad X_1 X_2 X_3
\]
Problem 1: Rigid Point Matching

Propagation in the Junction Tree:

\[ X_1 \ X_2 \ X_3 \ X_4 \]
\[ X_1 \ X_2 \ X_3 \ X_5 \]
\[ X_1 \ X_2 \ X_3 \ X_{T-1} \]
\[ X_1 \ X_2 \ X_3 \ X_T \]

\[ X_1 \ X_2 \ X_3 \]
\[ X_1 \ X_2 \ X_3 \]
Problem 1: Rigid Point Matching

Results over 300 trials, 10 node graphs

Fraction of correct correspondences

JT
GA
PRL
SB

std: position jitter
Problem 1: Rigid Point Matching

Results over 300 trials, 40 node graphs

Fraction of correct correspondences

std: position jitter

JT
GA
PRL
SB
Problem 1: Rigid Point Matching

std = 1, T = 10, varying S

Fraction of correct correspondences

Size of the codomain pattern (S)

Jitter = 1

X

Y
Problem 1: Rigid Point Matching

std = 4, T = 10, varying S

Fraction of correct correspondences

Size of the codomain pattern (S)

Jitter = 4

X Y
Problem 1: Rigid Point Matching

Matching 25 X 30 points in 'house' database

Separation between frames
Average correct correspondence
Matching 25 X 30 points in 'house' database
JT
GA
PRL
Problem 1: Rigid Point Matching

Matching 30 X 30 points in 'house' database

Separation between frames

Average correct correspondence

JT
GA
PRL
SB
Problem 2: Attributed Graph Matching
Problem 2: Attributed Graph Matching

Example: matching road networks
Problem 2: Attributed Graph Matching
Problem 2: Attributed Graph Matching
Problem 2: Attributed Graph Matching

- Typical unary attribute: length
- Typical pairwise attributes: relative angle, relative distance of centroids
Problem 2: Attributed Graph Matching
Problem 2: Attributed Graph Matching

Two angles uniquely determine the third
Problem 2: Attributed Graph Matching
Problem 2: Attributed Graph Matching
Problem 2: Attributed Graph Matching
For matching straight lines, a 3-tree model is also optimal in the noiseless case (because it implicitly encodes all distances \textit{and} all angles)
Size of the codomain pattern (S)
Fraction of correct correspondences
T = 10, std = 2, H kernel
JT4
JT3
JT2
PRL
Problem 3: Invariant Point Matching
Problem 3: Invariant Point Matching

- Key idea: Junction Tree agreement

- By varying $N$ from 1 to 4, one can implement translation, similarity, affine or projective invariance
Problem 3: Invariant Point Matching

Translation in the plane

A map \( P_1 = (x, y) \rightarrow p_1 = (x', y') \) **uniquely** defines a translation in the plane:

\[
x' = x + x_0 , \quad y' = y + y_0 \rightarrow 2 \text{ equations, 2 unknowns}
\]
Problem 3: Invariant Point Matching

Translation in the plane

A map $P_1 = (x, y) \rightarrow p_1 = (x', y')$ **uniquely** defines a translation in the plane:

\[ x' = x + x_0 \quad , \quad y' = y + y_0 \rightarrow \text{2 equations, 2 unknowns} \]

\[ \psi(X_1, X_i) \text{ is maximal for } M_1 = M_i \]

\[ [x_0^1 \quad y_0^1] = [x_0^2 \quad y_0^2] \]
Problem 3: Invariant Point Matching

Translation in the plane

A map \( P_1 = (x, y) \rightarrow p_1 = (x', y') \) uniquely defines a translation in the plane:

\[
x' = x + x_0, \quad y' = y + y_0 \rightarrow \text{2 equations, 2 unknowns}
\]

\[\psi(X_1, X_i)\text{ is maximal for } M_1 = M_i\]

\[
[x_0^1 y_0^1] = [x_0^3 y_0^3]
\]
Problem 3: Invariant Point Matching

Translation in the plane

A map $P_1 = (x, y) \rightarrow p_1 = (x', y')$ uniquely defines a translation in the plane:

$$x' = x + x_0, \quad y' = y + y_0 \rightarrow 2 \text{ equations, 2 unknowns}$$

$\psi(X_1, X_i)$ is maximal for $M_1 = M_i$

$$[x_0^1 y_0^1] = [x_0^{T-1} y_0^{T-1}]$$
Problem 3: Invariant Point Matching

Translation in the plane

A map $P_1 = (x, y) \rightarrow p_1 = (x', y')$ uniquely defines a translation in the plane:

$$x' = x + x_0, \quad y' = y + y_0 \rightarrow 2 \text{ equations, 2 unknowns}$$

\[ 
\begin{align*}
\psi(X_1, X_i) \text{ is maximal for } M_1 = M_i \\
[x_0^1 \ y_0^1] = [x_0^T \ y_0^T] \\
\text{Junction Tree consistency guarantees } M_1 = M_2 = \cdots = M_T
\end{align*}
\]
Problem 3: Invariant Point Matching

**Similarity transformation in the plane (rigid + scale)**

Similarly, a pairwise map:

\[ P_1 = (x_1, y_1) \rightarrow p_1 = (x'_1, y'_1) \]

\[ P_2 = (x_2, y_2) \rightarrow p_2 = (x'_2, y'_2) \]

**uniquely** defines a similarity transformation in the plane (4 eq. and 4 unknowns)
Problem 3: Invariant Point Matching

**Similarity transformation in the plane (rigid + scale)**

Similarly, a pairwise map:

\[ P_1 = (x_1, y_1) \rightarrow p_1 = (x'_1, y'_1) \]

\[ P_2 = (x_2, y_2) \rightarrow p_2 = (x'_2, y'_2) \]

**uniquely** defines a similarity transformation in the plane (4 eq. and 4 unknowns)

\[ \psi(X_1, X_2, X_i) \text{ is maximal for } M_{12} = M_{1i} = M_{2i} \]
Problem 3: Invariant Point Matching

Similarity transformation in the plane (rigid + scale)

Similarly, a pairwise map:

\[ P_1 = (x_1, y_1) \rightarrow p_1 = (x'_1, y'_1) \]
\[ P_2 = (x_2, y_2) \rightarrow p_2 = (x'_2, y'_2) \]

uniquely defines a similarity transformation in the plane (4 eq. and 4 unknowns)

\[
\psi(X_1, X_2, X_i) \text{ is maximal for } M_{12} = M_{1i} = M_{2i}
\]
Problem 3: Invariant Point Matching

**Similarity transformation in the plane (rigid + scale)**

Similarly, a pairwise map:

\[
P_1 = (x_1, y_1) \rightarrow p_1 = (x'_1, y'_1)
\]

\[
P_2 = (x_2, y_2) \rightarrow p_2 = (x'_2, y'_2)
\]

uniquely defines a similarity transformation in the plane (4 eq. and 4 unknowns)

\[
\psi(X_1, X_2, X_i) \text{ is maximal for } M_{12} = M_{1i} = M_{2i}
\]
Problem 3: Invariant Point Matching

**Similarity transformation in the plane (rigid + scale)**

Similarly, a pairwise map:

\[ P_1 = (x_1, y_1) \rightarrow p_1 = (x'_1, y'_1) \]

\[ P_2 = (x_2, y_2) \rightarrow p_2 = (x'_2, y'_2) \]

**uniquely** defines a similarity transformation in the plane (4 eq. and 4 unknowns)

\[ \text{Junction Tree consistency guarantees } M_{12} = M_{1i} = M_{2i} \]

\[ \psi(X_1, X_2, X_i) \text{ is maximal for } M_{12} = M_{13} = M_{23} = \cdots = M_{2T} \]
Problem 3: Invariant Point Matching

Affine transformation in the plane

Similarly, a 3-wise map

\[ P_1 = (x_1, y_1) \rightarrow p_1 = (x'_1, y'_1) \]
\[ P_2 = (x_2, y_2) \rightarrow p_2 = (x'_2, y'_2) \]
\[ P_3 = (x_3, y_3) \rightarrow p_3 = (x'_3, y'_3) \]

uniquely defines an affine transformation in the plane (6 eq. and 6 unknowns)
Problem 3: Invariant Point Matching

Projective transformation in the plane

Similarly, a 4-wise map

\[ P_1 = (x_1, y_1) \rightarrow p_1 = (x'_1, y'_1) \]
\[ P_2 = (x_2, y_2) \rightarrow p_2 = (x'_2, y'_2) \]
\[ P_3 = (x_3, y_3) \rightarrow p_3 = (x'_3, y'_3) \]
\[ P_4 = (x_4, y_4) \rightarrow p_4 = (x'_4, y'_4) \]

uniquely defines a projective transformation in the plane (8 eq. and 8 unknowns)
Problem 3: Invariant Point Matching
Problem 3: Invariant Point Matching
Problem 3: Invariant Point Matching
Problem 3: Invariant Point Matching
Final Comments: Advantages and Limitations

**Pros**

- Extremely robust to increasing pattern sizes
- Extremely robust to increasing size differences between patterns
- Rigid body version is very robust to noise
- No initialization, no iterative process, fully deterministic, global optimization

**Cons**

- Affine and Projective are much more sensitive to noise
- Current version is not robust to outliers in the template pattern
Conclusions

• Formulate matching as inference in graphical models

• Propose tractable model approximations and show that in the noiseless limit they are equivalent to the complete formulation

• Result: polynomial-time optimal solutions for a set of structural pattern recognition problems