Studying the subcellular location space with bioimages and other data modalities

Luis Pedro Coelho

Carnegie Mellon University

September 22 2011
Note: All material related to unpublished work has been removed from this online version.
(Wikipedia)
The Data
The Data
The Data
The Data
The Data
The Data
The Data
First Task

Classification

Given labeled data, can we learn a classification model?
Pre-processing

Preprocessing is denoising, background correction, . . .

I am going to ignore that step.
Features

Feature Based Approach

- Represent the image by a small number of features.
- Proposed by Boland and Murphy, in 1998.
- Very successful.
A feature is *any number you can compute from the image*.

For a good features, you wish to simultaneously

1. Capture the important variations.
2. Disregard the important variations.

These are naturally problem dependent,

but *machine learning helps*. 
Typical Features

- Texture (Haralick, Gabor, . . .)
- Edginess, smoothness, . . .
- . . .

The literature is very vast.
Example Feature

\[
\begin{array}{cccccc}
12 & 6 & 5 & 4 & 3 & 5 \\
11 & 10 & 4 & 6 & 7 & 4 \\
4 & 5 & 3 & 10 & 8 & 9 \\
3 & 4 & 12 & 9 & 8 & 14 \\
7 & 12 & 10 & 8 & 11 & 13 \\
\end{array}
\]
<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>6</td>
<td>5</td>
<td>4</td>
<td>3</td>
<td>5</td>
</tr>
<tr>
<td>11</td>
<td>10</td>
<td>4</td>
<td>6</td>
<td>7</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>3</td>
<td>10</td>
<td>8</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>12</td>
<td>9</td>
<td>8</td>
<td>14</td>
</tr>
<tr>
<td>7</td>
<td>12</td>
<td>10</td>
<td>8</td>
<td>11</td>
<td>13</td>
</tr>
</tbody>
</table>
12  6  5  4  3  5  
11  10  4  6  7  4  
4   5  3  10  8  9  
3   4  12  9  8  14 
7   12 10  8  11 13
For each $3 \times 3$ region:
- Find the maximum and the minimum.
- Subtract the minimum from the maximum.
- You end up with a number per region (per pixel).
For each $3 \times 3$ region:
- Find the maximum and the minimum.
- Subtract the minimum from the maximum.
- You end up with a number per region (per pixel).

For an image level feature, average this number.
Algorithm

- For each $3 \times 3$ region:
  - Find the maximum and the minimum.
  - Subtract the minimum from the maximum.
  - You end up with a number per region (per pixel).

For an **image level feature**, average this number

1. What is this feature **sensitive** to?
2. What is this feature **invariant** to?
Classifiers

Simple classifier: if average of region is less than 10, call it sky.
Classifiers

Simple classifier: if average of region is less than 10, call it sky.
## Results

<table>
<thead>
<tr>
<th></th>
<th>Cyto</th>
<th>Cytosk</th>
<th>Lyso</th>
<th>PM</th>
<th>Mito</th>
<th>N</th>
<th>NN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cyto</td>
<td>115</td>
<td>10</td>
<td>3</td>
<td>15</td>
<td>8</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>Cytosk</td>
<td>14</td>
<td>147</td>
<td>3</td>
<td>2</td>
<td>30</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Lyso</td>
<td>3</td>
<td>1</td>
<td>14</td>
<td>0</td>
<td>50</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>PM</td>
<td>31</td>
<td>6</td>
<td>2</td>
<td>9</td>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Mito</td>
<td>22</td>
<td>30</td>
<td>15</td>
<td>0</td>
<td>126</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>N</td>
<td>25</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>219</td>
<td>9</td>
</tr>
<tr>
<td>NN</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>16</td>
<td>95</td>
</tr>
</tbody>
</table>
Other Problems

Other Typical Classification Problems

- Phenotype in a Genome-wide Screen
- Stem cell differentiation
- ...
Conclusions

- Computers can do very well at classification.
- Flexible tool if you have the training data.
Detecting Changes in Pattern

Which proteins change pattern when we stress the cells?

In collaboration with Marc Halterman @ Rochester
Positive Example
Negative Example
K-Nearest Neighbour Test

(Henze, 1988)

(T. Zhao et al., 2006)
K-Nearest Neighbour Test

(Henze, 1988)

(T. Zhao et al., 2006)
K-Nearest Neighbour Test

(Henze, 1988)

(T. Zhao et al., 2006)
K-Nearest Neighbour Test

(Henze, 1988)

(T. Zhao et al., 2006)
K-Nearest Neighbour Test

(Henze, 1988)

(T. Zhao et al., 2006)
K-Nearest Neighbour Test

(Henze, 1988)

(T. Zhao et al., 2006)
K-Nearest Neighbour Test

(Henze, 1988)

(T. Zhao et al., 2006)
This is a simple example, but handling heterogeneity is an important area of research.
Can you follow the cells in a movie?
Example
Example
Techniques

Pipeline Approach:

- (Pre-process)
- Detect/Segment
- Link
Example
Greedy Linking Approach

**Linking Problem**

Given a set of points in one frame $\{x_i\}_i$ and points in another frame $\{y_i\}_i$, how do we link them together?

---

Marriage Problem

Given a bipartite graph $M \cup F$ with weights $w_{mf}$ between the members, can we find a matching that maximises $\sum w_{mf}$? Yes, we can solve this in polynomial time using the Hungarian algorithm.
Greedy Linking Approach

Linking Problem
Given a set of points in one frame \( \{x_i\}_i \)
and points in another frame \( \{y_i\}_i \),
how do we link them together?

Marriage Problem
Given a bipartate graph \( M \cup F \) with weights \( w_{mf} \) between the members,
can we find a matching that maximises \( \sum w_{mf} \)?

Yes, we can solve this in polynomial time using the Hungarian algorithm.
Better Approaches

Hungarian Algorithm

- Simple, off-the-shelf, solution.
- Computationally efficient
- Not very robust to noisy data

How to do better

- Take multiple time-steps into consideration.
- Active contours approaches combine segmentation and tracking.
Can we generate fake images that look like real images?
Generative Models

Why?

- Learn biology.
- Serve as input for simulation.
- Communication of image data.
- Test set generation.
Generative Models

How?

1. Define a model
2. Learn parameters from data
3. Generate images

(T. Zhao & Murphy, 2007)

(T. Peng et al., 2009)
Generative Models

How?

1. Define a model
2. Learn parameters from data
3. Generate images

(T. Zhao & Murphy, 2007)

(T. Peng et al., 2009)
Model Parameters

Number of filaments

Length Distribution

Elasticity

Generate Simulated Image

Update parameters as necessary

Compare with Real Image

Output: Optimal Model Parameters

(Shariff et al., 2009, 2010)
Two Images

(A. Shariff et al., 2009)
Previously reported methods work well for simple classes, like “endosomes” or “mitochondria.”
Previously reported methods work well for simple classes, like “endosomes” or “mitochondria.” What if a protein is present in both endosomes and mitochondria?
Mixture Pattern Example
Mixture Pattern Example
Mixture Pattern Example
Supervised Unmixing Problem

Given examples of pure patterns and a mixed pattern, can we identify how much each pure pattern contributes to the mixture?
Given examples of pure patterns and a mixed pattern, can we identify how much each pure pattern contributes to the mixture? Using an object-based approach, we can solve this.

(T. Zhao et al., 2005)
(T. Peng, G. Bonami et al., 2010)
Unsupervised Unmixing Problem

What if we don’t know the pure patterns?
Unsupervised Unmixing Problem

What if we don’t know the pure patterns?

Given a collection of untagged images, can we identify the pure and mixed patterns?
\[ \min \sum_i \| X_i - Bc \| \]

where \( B \) is the basis, \( c \) are the coefficients, and the problem subject to the constraint that the elements come from the dataset.
Results: Mixing Bases

(Coelho et al., 2010)
Results: Mixing Fractions

Correlation: 88%

Correlation: 91%

(Coelho et al., 2010)
Results: Mixing Fractions

Correlation: 88%

(Coelho et al., 2010)
Results: Mixing Fractions

Correlation:
88%

Correlation:
91%

(Coelho et al., 2010)
Pattern Unmixing works both in supervised and unsupervised modes.
Conclusions

1. Intersection between computer vision and biology.
2. Basic problems are solved with machine learning.
3. Problems from image processing
4. Specific problems to this area:
   - generative (or indirect) modeling
   - pattern unmixing
Areas of Active and Future Research

1. New modalities.
2. Computational efficiency.
3. Robust algorithms (less parameter tuning).
4. Post-pipeline algorithms.
5. Integrating human and computer computation.
6. Integrating different data modalities.
7. Finding needles in haystacks.
8. Machine learning with very noisy data, but lots of it (web community is going similar things, what can we borrow?)
10. Unsupervised or semi-unsupervised methods.
Acknowledgements

- Robert F. Murphy
- Tao Peng
- Aabid Shariff
- Ting Zhao
- Estelle Glory-Afshar
- Murphy Lab
Thank you.