Automated Interpretation of Subcellular Patterns in Microscope Images: Bioimage Informatics for Systems Biology

Estelle Glory (eglory@cmu.edu)
Murphy Group - Center for Bioimage Informatics
Carnegie Mellon University
Eukaryotic cells have many parts
Protein localization

- The sequence of each protein determines where it is localized in cells.
- Subsequences ("motifs") within a protein’s sequence are responsible for targeting it to one (or more) locations (structures/organelles).
Open questions

- How many distinct locations can proteins be found in? What are they?
- How many distinct motifs direct proteins to those locations? What are they?
The set of proteins expressed in a given cell type or tissue is called its proteome.

Proteomics projects:
- sequence
- structure
- activity
- partners
- location
All systems biology must be data driven

Key to progress
- identification of aspect that needs to be analyzed “ome-wide”
- development of assays and automated analysis approaches

Systems biology needs systematic information on high-resolution subcellular location
- Eventually, for every expressed protein for all cell types under all conditions

Providing this information is the goal of Location Proteomics
Automated Interpretation

- Traditional analysis of fluorescence microscope images has occurred by visual inspection
- Our goal over the past ten years has been to automate the interpretation, to yield better
  - Objectivity
  - Sensitivity
  - Reproducibility
Supervised Learning of High-Resolution Subcellular Location Patterns
The goal: Learn to recognize all major subcellular patterns

<table>
<thead>
<tr>
<th>ER</th>
<th>giantin</th>
<th>gpp130</th>
</tr>
</thead>
<tbody>
<tr>
<td>LAMP</td>
<td>Mito</td>
<td>Nucleolin</td>
</tr>
<tr>
<td>Actin</td>
<td>TfR</td>
<td>Tubulin</td>
</tr>
<tr>
<td>DNA</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2D Images of HeLa cells
The Challenge

- Pixel-by-pixel or region-by-region matching will not work for cell patterns because different cells have different shapes, sizes, orientations.
- Organelles/structures within cells are not found in fixed locations.
- Instead, describe each image numerically and compare the descriptors.
1. Create sets of images showing the location of many different proteins (each set defines one class of pattern)

2. Reduce each image to a set of numerical values ("features") that are insensitive to position and rotation of the cell

3. Use machine learning methods to “learn” how to distinguish each class using the features

Boland et al 1996; 1997; 1998; Boland & Murphy 2001; Huang & Murphy 2004
Example of classification using Morphological Features

<table>
<thead>
<tr>
<th>ER</th>
<th>Nucleoli</th>
</tr>
</thead>
<tbody>
<tr>
<td>108</td>
<td>6</td>
</tr>
<tr>
<td>83</td>
<td>232</td>
</tr>
<tr>
<td>31</td>
<td>4</td>
</tr>
</tbody>
</table>

- # of objects
- Average size of objects
- Average distance to COF

Any of these features could be used to distinguish these two classes.
Acquisition considerations

- Resolution defined as ability to distinguish two “point-sources”
- Maximal resolution in x-y plane given by Rayleigh (or Abbe) limit
  \[ \frac{1.22\lambda}{2\text{NA}} \]
  where \( \lambda \) is wavelength of emitted light and NA is the numerical aperture of the objective; 244 nm for 520 nm light and 1.3 NA
- Sampling theorem (Nyquist) says maximum information can be obtained if we sample at twice the maximum frequency present in a sample
- Try to achieve Nyquist Sampling at Rayleigh limit
Acquisition considerations

- Maintain low cell density if single cell measurements desired
- Control acquisition variables
  - Select (initial) focal plane consistently
  - Select fields consistently (at least one full cell per field)
  - Maintain constant camera gain, exposure time, number of slices
  - Select interphase cells or ensure sampling of cell cycle
Acquisition considerations

- Collect sufficient images per condition
  - For classifier training or set comparison, more than number of features
  - For classification or clustering, based on confidence level desired
- Collect reference images if possible (DNA, membrane)
Annotation considerations

- Maintain adequate records of all experimental settings
- Organize images by cell type/probe/condition
Preprocessing

- Correction for/Removal of camera defects
- Background correction
- Autofluorescence correction
- Illumination correction
- Deconvolution
2D slices (from bottom to top) for cell labeled for transferrin receptor (primarily in endosomes)
3D HeLa

- 2D slices (from bottom to top) for cell labeled for giantin (primarily in Golgi)
3D HeLa

- 2D slices (from bottom to top) for cell labeled for **tubulin** (major constituent of microtubules)
Single cell segmentation approaches

- Voronoi
- Watershed
- Seeded Watershed
- Level Set Methods
- Graphical Models
Voronoi diagram

Given a set of seeds, draw vertices and edges such that each seed is enclosed in a single polygon where each edge is equidistant from the seeds on either side.
Voronoi Segmentation Process

- Threshold DNA image (downsample?)
- Find the objects in the image
- Find the centers of the objects
- Use as seeds to generate Voronoi diagram
- Create a mask for each region in the Voronoi diagram
- Remove regions whose object that does not have intensity/size/shape of nucleus
Original DNA image
After thresholding and removing small objects
After triangulation
After removing edge cells and filtering
Final regions masked onto original image
Watershed Segmentation

- Intensity of an image ~ elevation in a landscape
  - Flood from minima
  - Prevent merging of “catchment basins”
  - Watershed borders built at contacts between basins

http://www.ctic.purdue.edu/KYW/glossary/whatisaws.html
Watershed Segmentation

- If starting image has intensity centered on the cells (e.g., DNA) that you want to segment, invert image so that bright objects are the sources.
- If starting image has intensity centered on the boundary between the cells (e.g., plasma membrane protein), don’t invert so that boundary runs along high intensity.
Seeded Watershed Segmentation

- Drawback is that the number of regions may not correspond to the number of cells.
- Seeded watershed allows water to rise only from predefined sources (seeds).
- If DNA image available, can use same approach to generate these seeds as for Voronoi segmentation.
- Can use seeds from DNA image but use total protein image for watershed segmentation.
Seeded Watershed Segmentation

Original image

Seeds and boundary

Applied directly to protein image (no DNA image)

Note non-linear boundaries
Level Set Methods

- Level set function \( \phi(x,y,t) \)
  - Positive inside the contour (mountain)
  - Negative outside the contour (valley)
  - Zero on the contour, \( C \) embedded at its zero level (sea level)

http://ranger.uta.edu/~alp/personal/travImageGallery.htm
Graphical Model Methods

Assumptions

- Two classes of pixels: those part of a cell or part of the background
- Each pixel is likely to be the same class as its neighbors
- Have information about where cells are likely to be and where boundaries (edges) are likely to be
- Probability that two pixels are same class related to probability that there is an edge between them
1. Start with initial DNA and edge potential

2. Run 1\textsuperscript{st} believe propagation (BP), separate foreground and background. Pick the most confidence foreground pixel $p$, set its DNA potential high

3. Run 2\textsuperscript{nd} BP, assign the pixels with the same class of $p$ to be \textit{segmented\_cell1}, then set these pixels to be background

4. Pick the most confident foreground pixel, Run BP, find another cell, and iterate....

5. Iteration stops when the segmented cell is too small

6. The resulting masks
Feature extraction
Morphological Features - Thresholding

- Morphological features require some method for defining objects
- Most common approach is global thresholding
- Methods exist for automatically choosing a global threshold (e.g., Riddler-Calvard method)
Ridler-Calvard Method

- Find threshold that is equidistant from the average intensity of pixels below and above it
Ridler-Calvard Method

Blue line shows histogram of intensities, green lines show average to left and right of red line, red line shows midpoint between them or the RC threshold
Ridler-Calvard Method

original

thresholded
Object finding

- After choice of threshold, define objects as sets of touching pixels that are above threshold
## 2D Features

### Morphological Features

<table>
<thead>
<tr>
<th>SLF No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLF1.1</td>
<td>The number of fluorescent objects in the image</td>
</tr>
<tr>
<td>SLF1.2</td>
<td>The Euler number of the image</td>
</tr>
<tr>
<td>SLF1.3</td>
<td>The average number of above-threshold pixels per object</td>
</tr>
<tr>
<td>SLF1.4</td>
<td>The variance of the number of above-threshold pixels per object</td>
</tr>
<tr>
<td>SLF1.5</td>
<td>The ratio of the size of the largest object to the smallest object</td>
</tr>
<tr>
<td>SLF1.6</td>
<td>The average object distance to the cellular center of fluorescence(COF)</td>
</tr>
<tr>
<td>SLF1.7</td>
<td>The variance of object distances from the COF</td>
</tr>
<tr>
<td>SLF1.8</td>
<td>The ratio of the largest to the smallest object to COF distance</td>
</tr>
</tbody>
</table>
## 2D Features

### DNA Features

**DNA features (objects relative to DNA reference)**

<table>
<thead>
<tr>
<th>SLF No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLF2.17</td>
<td>The average object distance from the COF of the DNA image</td>
</tr>
<tr>
<td>SLF2.18</td>
<td>The variance of object distances from the DNA COF</td>
</tr>
<tr>
<td>SLF2.19</td>
<td>The ratio of the largest to the smallest object to DNA COF distance</td>
</tr>
<tr>
<td>SLF2.20</td>
<td>The distance between the protein COF and the DNA COF</td>
</tr>
<tr>
<td>SLF2.21</td>
<td>The ratio of the area occupied by protein to that occupied by DNA</td>
</tr>
<tr>
<td>SLF2.22</td>
<td>The fraction of the protein fluorescence that co-localizes with DNA</td>
</tr>
</tbody>
</table>
## 2D Features

### Skeleton Features

<table>
<thead>
<tr>
<th>SLF No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLF7.80</td>
<td>The average length of the morphological skeleton of objects</td>
</tr>
<tr>
<td>SLF7.81</td>
<td>The ratio of object skeleton length to the area of the convex hull of the skeleton, averaged over all objects</td>
</tr>
<tr>
<td>SLF7.82</td>
<td>The fraction of object pixels contained within the skeleton</td>
</tr>
<tr>
<td>SLF7.83</td>
<td>The fraction of object fluorescence contained within the skeleton</td>
</tr>
<tr>
<td>SLF7.84</td>
<td>The ratio of the number of branch points in the skeleton to the length of skeleton</td>
</tr>
</tbody>
</table>
Illustration – Skeleton
## 2D Features

### Edge Features

<table>
<thead>
<tr>
<th>SLF No.</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SLF1.9</td>
<td>The fraction of the non-zero pixels that are along an edge</td>
</tr>
<tr>
<td>SLF1.10</td>
<td>Measure of edge gradient intensity homogeneity</td>
</tr>
<tr>
<td>SLF1.11</td>
<td>Measure of edge direction homogeneity 1</td>
</tr>
<tr>
<td>SLF1.12</td>
<td>Measure of edge direction homogeneity 2</td>
</tr>
<tr>
<td>SLF1.13</td>
<td>Measure of edge direction difference</td>
</tr>
</tbody>
</table>
2D Features
Haralick Texture Features
(SLF7.66-7.78)

- Correlations of adjacent pixels in gray level images
- Start by calculating co-occurrence matrix $P$:
  N by N matrix, $N=$number of gray level.
  Element $P(i,j)$ is the probability of a pixel with value $i$
  being adjacent to a pixel with value $j$
- Four directions in which a pixel can be adjacent
- Each direction considered separately and then
  features averaged across all directions
Co-occurrence Matrices

Example image with 4 gray levels

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>

Carnegie Mellon
Pixel Resolution and Gray Levels

- Texture features are influenced by the number of gray levels and pixel resolution of the image
- Optimization for each image dataset required
- Alternatively, features can be calculated for many resolutions
2Dt or 3Dt Features

Temporal Texture Features

- **Haralick texture features** describe the correlation in intensity of pixels that are next to each other in **space**.
  - These have been valuable for classifying static patterns.

- **Temporal texture features** describe the correlation in intensity of pixels in the same position in images next to each other over **time**.
Temporal Textures based on Co-occurrence Matrix

- Temporal co-occurrence matrix $P$: $N_{\text{level}} \times N_{\text{level}}$ matrix, Element $P[i, j]$ is the probability that a pixel with value $i$ has value $j$ in the next image (time point).
- Thirteen statistics calculated on $P$ are used as features.
<table>
<thead>
<tr>
<th></th>
<th>Image at t0</th>
<th></th>
<th>Image at t1</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>2</td>
<td>2</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>4</td>
</tr>
</tbody>
</table>

**Temporal co-occurrence matrix (for image that does not change)**

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>9</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>7</td>
</tr>
</tbody>
</table>
### Temporal Co-occurrence Matrix (for Image that Changes)

**Image at t0**

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>

**Image at t1**

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>1</td>
<td>4</td>
<td>4</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>2</td>
<td>1</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>

**Temporal co-occurrence matrix**

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>5</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>
Implementation of Temporal Texture Features

- Compare image pairs with different time interval, compute 13 temporal texture features for each pair.

- Use the average and variance of features in each kind of time interval, yields $13 \times 5 \times 2 = 130$ features
Machine Learning - Classification Methods
Simple two class problem
k-Nearest Neighbor (kNN)

- In feature space, training examples are

Feature #1 (e.g., 'area')

Feature #2 (e.g., roundness)
We want to label ‘?’

Feature #2 (e.g., roundness)

Feature #1 (e.g., ‘area’)

k-Nearest Neighbor (kNN)
**k-Nearest Neighbor (kNN)**

- Find k nearest neighbors and vote

Feature #2 (e.g., roundness)

Feature #1 (e.g., ‘area’)

So we label it +

for k=3, nearest neighbors are
Decision trees

- Again we want to label ‘?’

Feature #2 (e.g., roundness)

Feature #1 (e.g., ‘area’)

Slide courtesy of Christos Faloutsos
Decision trees

- so we build a decision tree:

Feature #1 (e.g., ‘area’)

Feature #2 (e.g., roundness)

40

50

Slide courtesy of Christos Faloutsos
Decision trees

- so we build a decision tree:

Slide courtesy of Christos Faloutsos
Decision trees

- Goal: split address space into (almost) homogeneous regions

Slide courtesy of Christos Faloutsos
Support vector machines

- Again we want to label ‘?’

Feature #1 (e.g., ‘area’)

Feature #2 (e.g., roundness)

Slide courtesy of Christos Faloutsos
Support Vector Machines (SVMs)

- Use single linear separator??

Slide courtesy of Christos Faloutsos
Support Vector Machines (SVMs)

- Use single linear separator??

Slide courtesy of Christos Faloutsos
Support Vector Machines (SVMs)

- Use single linear separator??

Slide courtesy of Christos Faloutsos
Support Vector Machines (SVMs)

- Use single linear separator??

round.

area

Slide courtesy of Christos Faloutsos
Support Vector Machines (SVMs)

- Use single linear separator??

Slide courtesy of Christos Faloutsos
Support Vector Machines (SVMs)

- we want to label ‘?’ - linear separator??
- A: the one with the widest corridor!

Slide courtesy of Christos Faloutsos
Support Vector Machines (SVMs)

- we want to label ‘?’ - linear separator??
- A: the one with the widest corridor!

Slide courtesy of Christos Faloutsos
Cross-Validation

- If we train a classifier to minimize error on a set of data, have no ability to generalize error that will be seen on new dataset
- To calculate *generalizable* accuracy, we use \( n \)-fold cross-validation
- Divide images into \( n \) sets, train using \( n-1 \) of them and test on the remaining set
- Repeat until each set is used as test set and average results across all trials
Describing classifier errors

- For multi-class classifiers, typically report
  - **Accuracy** = \( \frac{\text{# test images correctly classified}}{\text{# test images}} \)

- For binary classifiers (positive or negative), define
  - **TP** = true positives, **FP** = false positives
  - **TN** = true negatives, **FN** = false negatives
  - **Recall** = \( \frac{TP}{TP + FN} \)
  - **Precision** = \( \frac{TP}{TP + FP} \)
  - **F-measure** = \( \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \)
2D Classification Results

<table>
<thead>
<tr>
<th>True Class</th>
<th>DNA</th>
<th>ER</th>
<th>Gia</th>
<th>Gpp</th>
<th>Lam</th>
<th>Mit</th>
<th>Nuc</th>
<th>Act</th>
<th>TfR</th>
<th>Tub</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNA</td>
<td>99</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>ER</td>
<td>0</td>
<td>97</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Gia</td>
<td>0</td>
<td>0</td>
<td>91</td>
<td>7</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>Gpp</td>
<td>0</td>
<td>0</td>
<td>14</td>
<td>82</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Lam</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>88</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>Mit</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>92</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Nuc</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>99</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Act</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TfR</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>12</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>81</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Tub</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>95</td>
</tr>
</tbody>
</table>

Overall accuracy = 92%

Murphy et al 2000; Boland & Murphy 2001; Huang & Murphy 2004
<table>
<thead>
<tr>
<th>True Class</th>
<th>DNA</th>
<th>ER</th>
<th>Gia</th>
<th>Gpp</th>
<th>Lam</th>
<th>Mit</th>
<th>Nuc</th>
<th>Act</th>
<th>TfR</th>
<th>Tub</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNA</td>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>ER</td>
<td>0</td>
<td>90</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Gia</td>
<td>0</td>
<td>0</td>
<td>56</td>
<td>36</td>
<td>3</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Gpp</td>
<td>0</td>
<td>0</td>
<td>54</td>
<td>33</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>Lam</td>
<td>0</td>
<td>0</td>
<td>6</td>
<td>0</td>
<td>73</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>20</td>
<td>0</td>
</tr>
<tr>
<td>Mit</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>96</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>Nuc</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Act</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TfR</td>
<td>0</td>
<td>13</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>83</td>
<td>0</td>
</tr>
<tr>
<td>Tub</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>93</td>
</tr>
</tbody>
</table>

Overall accuracy = 83%
Computer vs. Human

Computer Accuracy

Carnegie Mellon
3D HeLa cell images

Images collected using facilities at the Center for Biologic Imaging courtesy of Simon Watkins
### 3D Classification Results

<table>
<thead>
<tr>
<th>True Class</th>
<th>DNA</th>
<th>ER</th>
<th>Gia</th>
<th>Gpp</th>
<th>Lam</th>
<th>Mit</th>
<th>Nuc</th>
<th>Act</th>
<th>TfR</th>
<th>Tub</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNA</td>
<td>98</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>ER</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Gia</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Gpp</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>96</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Lam</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>96</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Mit</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>96</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Nuc</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td>0</td>
</tr>
<tr>
<td>Act</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>100</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TfR</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>96</td>
<td>2</td>
</tr>
<tr>
<td>Tub</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>98</td>
</tr>
</tbody>
</table>

**Overall accuracy = 98%**
Conclusions (1996-2004)

- Automated classification of subcellular patterns possible without colocalization
- Accuracy better than visual examination
  - Similar for basic patterns
  - Better for similar patterns
- 3D images give better accuracy than 2D
- >> SLFs capture essence of patterns
Unsupervised Learning to Identify High-Resolution Protein Patterns
Location Proteomics

- **Tag** many proteins
  - We have used **CD-tagging** (developed by Jonathan Jarvik and Peter Berget): Infect population of cells with a retrovirus carrying DNA sequence that will “tag” in a random gene
Principles of CD-Tagging (Jarvik & Berget) (CD = Central Dogma)

Genomic DNA + CD-cassette
Tagged DNA
Tagged mRNA
Tagged Protein

Tag (Epitope)
Location Proteomics

- **Tag** many proteins
  - We have used **CD-tagging** (developed by Jonathan Jarvik and Peter Berget): Infect population of cells with a retrovirus carrying DNA sequence that will “tag” in a random gene
  - Isolate separate **clones**, each of which produces express one tagged protein
  - Use RT-PCR to identify **tagged gene** in each clone
  - Collect **many live cell images** for each clone using spinning disk confocal fluorescence microscopy

Jarvik et al 2002

Carnegie Mellon
What Now?

Group ~90 tagged clones by pattern

Carnegie Mellon
Solution: Group them automatically

How?

Features can be used to measure similarity of protein patterns

This allows us for the first time to create a systematic, objective, framework for describing subcellular locations: a Subcellular Location Tree

Start by grouping two proteins whose patterns are most similar, keep adding branches for less and less similar patterns

Chen et al 2003; Chen and Murphy 2005
### Nucleolar Proteins

<table>
<thead>
<tr>
<th>Protein</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nucleoplasmin</td>
<td>Nucleolus</td>
</tr>
<tr>
<td>Cytokeratin</td>
<td>Nucleolus</td>
</tr>
<tr>
<td>Nucleolin</td>
<td>Nucleolus</td>
</tr>
<tr>
<td>Histone</td>
<td>Nucleolus</td>
</tr>
<tr>
<td>Ribosomal RNA</td>
<td>Nucleolus</td>
</tr>
</tbody>
</table>

**Z-scored Euclidean Distance**

- similarities between nucleolar proteins
- dendrogram showing relationships

**Images:**
- Low-magnification microscopy
- High-magnification microscopy
Punctate Nuclear Proteins
Predominantly Nuclear Proteins with Some Punctate Cytoplasmic Staining
Nuclear and Cytoplasmic Proteins with Some Punctate Staining
Generative Models for Subcellular Location Patterns
Need

- How do we communicate results of clustering patterns?
- Show all images from a given cluster?
  - Long download
  - No ability to generalize
- Proposal: Use generative models
LAMP2 pattern

Cell membrane
Nucleus
Protein
Synthesized Images

Lysosomes

Endosomes
Synthesized Images

Mitochondria

Nucleoli
Model Distribution

- Generative models provide better way of distributing what is known about “subcellular location families” (or other imaging results, such as illustrating change due to drug addition)
- Have initial XML design for capturing the models for distribution
- Have portable tool for generating images from the model
Generation Process

- Protein
- Cell Shape
- Nuclear Model
- XML
Generating Multiple Distributions for Simulations

Protein
Cell Shape
Nuclear Model

Simulation 1
Simulation 2
Simulation 3

Conclusions
Combining Models for Cell Simulations

- Protein 1
  - Cell Shape
  - Nuclear Model

- Protein 2
  - Cell Shape
  - Nuclear Model

- Protein 3
  - Cell Shape
  - Nuclear Model
  - XML

Shared Nuclear and Cell Shape

Simulation
The Protein Subcellular Location Image Database (PSLID)
PSLID: Protein Subcellular Location Image Database

- A publicly accessible image database at http://pslid.cbi.cmu.edu
  - Version 3 released February 2, 2007
  - 2D and 3D images (single cell regions defined)
  - Two cell types, HeLa and 3T3
  - Over 120,000 images/3000 unique fields/14,000 cells
  - 111 classes; 55 known proteins; 11 targeting mutants of a single protein
  - Programmatic search via URL
PSLID: Protein Subcellular Location Image Database

- A downloadable open source system for creating local databases
  - Version 3 of software released February 13, 2007
  - Focused on subcellular pattern analysis
  - SLF features integrated into database
  - Integrated comparison, classification, clustering tools
  - Designed for high-throughput microscopy
  - Interface to OME in the works
  - Large ITR project with UCSB for distributed system

Huang et al 2002; Huang et al 2007
PSLID stands for Protein Subcellular Localization Image Database. PSLID collects and structures 2-D through 5-D fluorescence microscope images, annotations, and derived features in a relational schema.

It is designed so that interpretations as well as annotations can be queried. The annotations in PSLID, composed of 44 linked tables with publicly available descriptions, provide a thorough description of sample preparation and fluorescence microscope imaging.

Image interpretation is achieved using Subcellular Location Features that have been shown to be capable of recognizing all major subcellular structures and of resolving patterns that cannot be distinguished by eye.

The fundamental unit of PSLID is an image set, which is simply a logical grouping of images. Image sets can be defined at the time of image loading, or they can be defined by searching for images that meet specified criteria (e.g., all images of "actin" or all images that are similar to a query image). They can also be created by analysis functions such as cluster analysis (e.g., the images in each cluster found by cluster analysis can be put into distinct sets).

Analysis capabilities that are incorporated in PSLID include:

- **Searching** for images by context (annotations) or content
- **Ranking images by typicality** within a set
  - e.g., to choose an image for presentation or publication
- **Ranking images by similarity** to one or more query images
  - "searching by image content" or "relevance feedback"
- **Comparing** two sets of images (hypothesis testing)
  - e.g., to determine whether a drug alters the distribution of a tagged protein
- **Training a classifier** to recognize subcellular patterns
- **Using a trained classifier** to assign images to pattern classes
  - e.g., assigning images to "positive" or "negative"
- **Clustering images** by their subcellular patterns
  - e.g., finding "subcellular location families" within a large set of images

You can go to the [Quick Start](#) page to see instructions for PSLID installation, image loading, and image analysis using PSLID.

The public PSLID database currently contains a number of large image collections. It can be accessed interactively or via queries embedded in URLs. We encourage the submission to PSLID of other image collections documenting the subcellular location of proteins to facilitate "one-stop" searching for information on subcellular patterns.

**Login**  **Public Access**  **Quick Start**
External search

## Search results for Image Type: 2D Static, Target: calponin-2

10 regions returned (30 regions shown) from the query.

View the summary of set temp8_710B35DB64C10A8CF219992B3A193B57.

Click 📅 beside a given image to retrieve similar images in the database.

<table>
<thead>
<tr>
<th>Image</th>
<th>Cell Name</th>
<th>Organism</th>
<th>Segmenter</th>
<th>Experiment</th>
<th>Protocol</th>
<th>Target</th>
<th>Microscopy &amp; Filter</th>
</tr>
</thead>
<tbody>
<tr>
<td>Region 68249</td>
<td>3T3</td>
<td>Mus musculus</td>
<td>External</td>
<td>Cyto039</td>
<td>GFP_Live</td>
<td>Calponin-2</td>
<td>Olympus IX500</td>
</tr>
<tr>
<td>Region 68280</td>
<td>3T3</td>
<td>Mus musculus</td>
<td>External</td>
<td>Cyto039</td>
<td>GFP_Live</td>
<td>Calponin-2</td>
<td>Olympus IX500</td>
</tr>
<tr>
<td>Region 68311</td>
<td>3T3</td>
<td>Mus musculus</td>
<td>External</td>
<td>Cyto039</td>
<td>GFP_Live</td>
<td>Calponin-2</td>
<td>Olympus IX500</td>
</tr>
<tr>
<td>Region 68342</td>
<td>3T3</td>
<td>Mus musculus</td>
<td>External</td>
<td>Cyto039</td>
<td>GFP_Live</td>
<td>Calponin-2</td>
<td>Olympus IX500</td>
</tr>
<tr>
<td>Region 68373</td>
<td>3T3</td>
<td>Mus musculus</td>
<td>External</td>
<td>Cyto039</td>
<td>GFP_Live</td>
<td>Calponin-2</td>
<td>Olympus IX500</td>
</tr>
<tr>
<td>Region 68404</td>
<td>3T3</td>
<td>Mus musculus</td>
<td>External</td>
<td>Cyto039</td>
<td>GFP_Live</td>
<td>Calponin-2</td>
<td>Olympus IX500</td>
</tr>
<tr>
<td>Region 68435</td>
<td>3T3</td>
<td>Mus musculus</td>
<td>External</td>
<td>Cyto039</td>
<td>GFP_Live</td>
<td>Calponin-2</td>
<td>Olympus IX500</td>
</tr>
</tbody>
</table>
Conclusions

- Methods well worked out for classifying and learning protein patterns - better than visual examination
- Temporal information improves discrimination
- Progress on decomposing complex patterns and synthesizing distributions
  - High-resolution, reliable data for bottom-up systems modeling
- Graphical models provide improved classification of single cells in fields (and potentially tissues)
  - New fast inference algorithm
- Image database integrated with interpretation tools (PSLID)
- Information extractor for online text and images (SLIF)
Acknowledgments

- **Students**
  - Dr. Michael Boland
  - Dr. Mia Markey (ugrad)
  - Gregory Porreca (ugrad)
  - Dr. Meel Velliste
  - Dr. Kai Huang
  - Dr. Xiang Chen
  - Dr. Ting Zhao
  - Dr. Juchang Hua
  - Dr. Yanhua Hu
  - Shann-Ching Chen

- **Funding**
  - NSF, NIH, Commonwealth of Pennsylvania

- **Collaborators/Consultants**
  - NSF ITR: B. S. Manjunath Ambuj Singh

---

Carnegie Mellon
Vision

- Full automation of
  - experiment design
  - adaptive acquisition
  - model-based image interpretation
- to generate biological knowledge from images in a form suitable for systems modeling
The Future of Subcellular Pattern Analysis
The problem

Cell Type (Order $10^2$)  Condition (Order $10^2$)

Plus: Time scale from subsecond to years
Other subcellular location projects

- O’Shea group - Yeast
  - GFP-tagged cDNAs
  - GFP and DNA images with some additional markers
- Pepperkok group - human (MCF7 cells)
  - GFP-tagged cDNAs
  - GFP and DNA images
- Uhlen group (Protein Atlas) - human
  - Immunohistochemistry with monospecific antibodies
  - DAB and hematoxylin images
  - Fixed tissues
- Schubert group (MELK technology)
  - Cycles of immunofluorescence, imaging and bleaching
  - Fixed tissues
- Teasdale group (Locate, Hela)
  - Immunofluorescence and GFP-tagged proteins
  - GFP and DNA images
How do we really analyze subcellular location?

- Classification and comparison good for focused questions but there are too many questions to ask.
- Scope of problem argues for cooperation on grand scale: Human Cytome Project?
- Need intelligent (optimized) data collection: probabilistic methods to integrate available data, make predictions and suggest experiments.
Welcome to the Molecular Biosensor and Imaging Center website.

Mission

To develop fluorescence detection technologies for biomedical research and NASA space exploration.

NIH Technology Center for Networks and Pathways

Alan Waggoner
What do we do

1. **Biology**: Pose a question about a biological system
2. **Acquisition**: Design strategy for collecting relevant information in the form of images of molecules, cells, organisms
3. **Signal Processing/Computer Science**: Find the answer through image processing and machine learning
4. **Scientific Computing**: Optimize computational performance for real-time applications and sharing

For more information: [http://www.cbi.cmu.edu](http://www.cbi.cmu.edu)
Collaborations with Bill Mohler, Ian Moraru, Les Loew, Paul Campagnola (U Conn)

Collaboration with Badri Roysam (RPI) and Sally Temple (Albany Med Coll), Stem Cell Patterning and FARSIGHT system

Collaboration with Dan Rines and Sumit Chanda (GNF San Diego) on high throughput location proteomics
Thank you!