University of California, Santa Barbara
Center for Bio-Image Informatics

2004 Summer Intern Program

University of California, Santa Barbara
Center for Bio-Image Informatics

2004 Summer Intern Program

The UCSB Center for Bio-Image Informatics provided a summer research fellowship program for undergraduate students in Biology and Information Technology. Ten undergraduate students from across the state participated in the internship. The Center's mission is to establish a searchable digital library for bio-molecular images and to develop new information processing technologies for a better understanding of complex biological processes at the cellular and molecular level. The Summer Undergraduate Research fellows interacted with faculty and graduate student mentors involved in this project while also attending various trainings designed to develop the specific skills necessary for success at the graduate level. The participants were involved in cutting edge research in biology, imaging, image processing, databases, machine learning, pattern recognition, and data mining.

Each intern completed a research project under the supervision of one of the laboratories associated with the program. An intern was paired with a mentor (a graduate student or postdoc) who was affiliated with his or her assigned laboratory. To help the mentors in their role, workshops were held on how to develop a research plan for the students, good mentoring practices, and teaching skills.

The interns also interacted with other students and faculty who were part the project. They attended a weekly workshop on communication skills, presentations covering both the project and general scientific topics, and weekly project meetings. Additionally, the students received guidance and information regarding opportunities in science and how to pursue graduate education.

The interns learned to write scientific abstracts and summaries of their work, designed and created posters for their project, and produced and presented PowerPoint presentations. The final abstracts were incorporated into the UCSB Summer Undergraduate Research Colloquium and their posters were displayed at this campus-wide event. The Bio-Image project held a separate Summer Intern Symposium where the students presented the work they accomplished to faculty and students. Faculty from California State University San Bernardino attended and the Vice-Chancellor for Graduate Studies spoke to the students.

We would like to extend our appreciation to Dr. Fiona Goodchild, Liu-Yen Kramer, and Trevor Hirst for their advice, support, and their willingness to accommodate the Bio-image Students with those in the California Nanosystems Institute. We would also like to acknowledge and thank the graduate mentors and faculty for providing the research opportunities for the interns and their supervision. Thanks in particular to Maura Jess for her continued support throughout the entire course of the program.

This summer program was supported by the National Science Foundation Information Technology research (ITR) program and Grant No. 0331697. The College of Engineering, the College of Letters and Science, and the Graduate Division provided additional funding.

B.S. Manjunath
Director, Center for Bio-Image Informatics

Mark Verardo
Education and Outreach Coordinator

University of California, Santa Barbara
Center for Bio-Image Informatics

2004 Summer Intern Program

"This gave me a taste of what graduate research could be like."

DATA HIDING IN BIOLGY IMAGES

David Wheland

University of California, Santa Barbara

Mentor: Ken Sullivan

Faculty Advisor: B.S. Manjunath

I started out my internship learning how to use Matlab, a program designed for matrix calculations. It has come into widespread use due to its versatile libraries that save programmers time by providing pre-made functions. I then studied mathematical transforms, such as the discrete Fourier and cosign transforms. This would become the foundation to data hiding. Transforms allow the pixel data of the image to be transformed to the frequency domain and decorrelated so that changes to the transform coefficients result in a less noticeable change to the image overall. I began working in Matlab on a simple quantization index modulation method, which is the basis of UCSB's data hiding scheme. Here is an example of how this method works with a step size of 2 (distance between the same bit value of 0 is 2): a transform coefficient in which data is to be hidden would be rounded to the nearest even number to represent a bit value of 0 or to the nearest odd number to represent a bit value of 1. JPEG standard could be used in the hiding technique to make the hidden data resistant to JPEG compression. From this I received the matrix specifying optimal delta sizes to use in hiding for certain transform coefficients.

Hiding data in biology images was more difficult than hiding in most images due to the large amount of black, low frequency space. Using this kind of space for hiding data makes the image appear noisy or blurry. Because of this, a unique scheme had to be developed for hiding data in biology images. After doing some tests with certain parameters, I ended up using a customized variation of the UCSB scheme in which only certain, less-influential transform coefficients are hidden in. The biology image scheme adjusted the threshold values used to decide whether to hide in a coefficient. It also modified the way in which the coefficients were rounded for the quantization/rounding part of the scheme. This proved to be a stark improvement over the original method, although it resulted less data being able to be hidden in the image. None the less, more than enough data was still able to be hidden (200 characters worth). The next step was to implement error correction code for the hidden data so that completely accurate data could be returned out from the image. To do this I used the repeat accumulator error correction code, which approaches the theoretical maximum error correction efficiency. Implementing this, however, required far more work than expected as the code's many parameters had to be finely tuned for use with biology images. This proved to have very different parameter values than many common images.

Having made serious headway in the data hiding project, I finished up my internship learning asp.net and developed a webpage demo for the data hiding work.

Data Hiding in Biology Images

David Wheland, UCSB

Kenneth Sullivan, Kaushal Solanki, B.S. Manjunath, Vision @ UCSB

Research Objectives

- Hide/Embed 200 bytes of data in biology-images.
- Minimize image distortion due to the hiding process.
- Errorlessly retrieve hidden data from images using error correction code.
- Automate entire process so that it runs in a timely fashion.

Approach

Broader Impacts

- Descriptive biology data "permanently" coupled with its respective image:
  - If you have the image, then you have its data.
  - Data is not easily estranged from its image, contrary to a database approach.
- Saves time: data base creation and management unnecessary

Significant Results

Adaptive Hiding

- **Adaptive Step:** For each transform coefficient check if its magnitude is above a certain threshold value. If so, proceed to hide data in it.
- **Quantization Hiding (Step Size = 2):**
- If the next bit to be hidden is a 1, round the coefficient to the closest odd number.
- If the next bit to be hidden is a 0, round the coefficient to the closest even number.

University of California, Santa Barbara
Center for Bio-Image Informatics
2004 Summer Intern Program

"I like the way that people have their individual work shown and [how they] interact with others."

CLUSTERING BIO-IMAGES BASED ON TEXTURE

Jie Jun Xu

California State University, Fresno

Mentor: Alphon Altinok

Faculty Advisor: Ken Rose

1. Introduction

Biological databases are becoming increasingly popular in part due to the large amount of images that are generated by various applications and the advances in computational technology. One of the typical application areas of such systems is a bio-images database. During the previous eight weeks, I worked on developing a clustering approach based on image texture for fast retrieval bio-images from a database.

The general methodology used was: (1) Extract texture features from images and construct a feature vector by using co-occurrence matrixes for each image (2) Feature vectors are clustered into groups by using k-mean algorithms. The mean of each cluster is selected as a representative for each group and will be used for narrowing down the entire search space. In this paper texture features and their extraction methods are discussed. Also, the clustering process will be presented. Finally, the results obtained from our experimentation will be presented.

2. Texture Extraction

Texture is defined as a function of the spatial variation in pixel intensities and characterized by the spatial distribution of gray levels in a neighborhood. It is one of the most important features used in recognizing patterns in an image. In my project, I chose one group of widely used texture features proposed by Haralick, which is based on the co-occurrence matrix. These features give a measurement of how often one gray value will appear in a specified spatial relationship to another gray value on the image.

The co-occurrence matrix **P** is defined as follows. The element **P**(i, j) of this matrix for a window is the number of times, divided by N, that gray-levels i and j occur in two pixels satisfy a specified spatial separation, where N is the number of pixels pairs contributing to **P**. From the co-occurrence matrix, a total of fourteen Haralick's statistical texture features can be computed. Below are some examples (6 out of 14).

Contrast

$$\sum n^2 \sum \sum (i-j)^2 P(i,j)$$

Sum of squares

$$\sum \sum (i-j)^2 P(i,j)$$

Variance

$$\mathrm{var}(\mathbf{P}_{x+y})$$

Inverse diff moment

$$\sum \sum [1+(i-j)^2]^{-1} P(i,j)$$

Sum average

$$\sum \sum i P_{x+y}(i)$$

Energy

$$\sum \sum P^2(i,j)$$

One important thing is instead of using one single co-occurrence matrix for an image, I tiled an image to a certain number of sub-images, and co-occurrence matrixes were applied to each of those. Then I extracted the Haralick's features from each sub-image. After all the computation, one image was represented as one feature vector. The dimension of the feature vector was equal to the number of features extracted from each sub-image times the number of sub-images. At the end, all image features in the database are store in a large matrix, and each row stands for one image.

3. Clustering

The k-means algorithm was used for clustering. It is an algorithm to cluster or group objects based on attributes/features into K number of groups. K is positive integer. The grouping is done in iterations, which minimizes the sum of squares of distance between data and the corresponding cluster centroid. In our case, each row as a whole was treated as data. After the clustering is done, the mean of each cluster is selected to be the representative of that cluster. As mentioned before, clustering is a convenient method for organizing a large set of data so that the retrieval of information can be made more efficiently. Cluster representatives may provide a very convenient summary of the database, and it will increase the retrieval speed significantly.

4. Result

Currently I have about a thousand images in my database, and I have extracted features and clustered them into ten groups by using k-means algorithm (K= 10). Also, image searching has been performed on my database. The results are shown below. Based on the results, I would conclude that the clustering is performing the image search. However, I believe that by deeper analyzing the program, further improvement can be made.

Clustering Bio-Images Based on Texture

JieJun Xu, CSU Fresno
Alphan Altinok CS, Kenneth Rose, ECE UCSB

Research Objective:

- Develop a clustering approach for fast retrieval bio-images from an image database.
- Address similarity metrics using different features to obtain multiple partitioning of the image space.

Impact:

- Unsupervised learning methods will enable the database to adapt as additional images are added.
- Improve the speed of searching.

Texture examples:

Gray level cooccurrence matrix:

The GLCM is a two dimensional dependency matrix which gives a measure of how often one gray value will appear in a specified spatial relationship to another gray value in the image.

2	1	2	0	1
0	2	1	1	2
0	1	2	2	0
1	2	2	0	1
2	0	1	0	1

⟹ i, j ⟹

d = (1,1)

0	2	2	0
2	1	2	1
2	3	2	2
0	1	2	

i, j

Haralick's features: (6 out of 14)

Contrast

$$\sum n^2 \sum\sum (i-j)^2 P(i,j)$$

Sum of squares

$$\sum\sum (i-j)^2 P(i,j)$$

Variance

$$\mathrm{var}(\mathbf{P}_{x+y})$$

Inverse diff moment

$$\sum\sum [1+(i-j)^2]^{-1} P(i,j)$$

Sum average

$$\sum\sum i P_{x+y}(i)$$

Energy

$$\sum\sum P^2(i,j)$$

GLCM

Clustering:

Hierarchical K-means algorithm.

Methodology:

Results:

- Find the cluster to which the query image belongs.
- Retrieve *n* closest images from the cluster and their distances to query image.

query *distance from query image*

Future Directions:

- Implement fuzzy k-means as images could belong to different clusters with various degrees at the same time.
- Experiment with other features than texture for alternative partitionings.
- Experiment with number of clusters to find optimal partitioning of image space.

"This eight week program was a great opportunity to get an authentic grasp on what graduate research is really like..."

SEARCH ENGINES FOR IMAGES OF RETINAS EXPOSED TO HYPEROXIC AND NOMOXIC CONDITIONS

Joriz De Guzman

California State University, San Bernardino

Mentor: Laura Boucheron

Faculty Advisor: B.S. Manjunath

During this summer, I was given the opportunity to work and be part of the Bio-Image Informatics research project with Dr. Manjunath. The following paragraphs will give a brief description of the motivation behind my projects and a detailed description of the software that I implemented.

First and foremost, the general goal for my project is to allow biologists to make necessary assumptions about feline retinas that were stored in two types of oxygen chambers. Those two chambers were normoxic (~21% oxygen) and hyperoxic (~70% oxygen). In a normoxic environment, photoreceptors in the retina do not retain their normal position. This means that when photoreceptors separate from the retina and experience cell death, they move in different directions. Once these photoreceptors reattach to the retina in an unordered fashion, vision quality is lost and also the ability to discriminate colors is decreased. Simply, abnormal photoreceptors lead to blindness. However, in a hyperoxic environment, photoreceptors stay in place and stay aligned so when they reattach to the retina, the degree of blindness is decreased. So, it is of interest to Biologists to visually see the difference between normoxic and hyperoxic images. This is related to my project because my goal was to implement a search engine that would allow its users to view the Bio-Images and also the Metadata (the information extracted from each of the images). The following paragraph will give a more in depth description of what the functionality of the search engine is and the method in which it was implemented.

The search engine currently has a database of 142 Bio-Images (both normoxic and hyperoxic images). The search engine utilizes four different criteria to execute searches. The four criteria are the Average Segment Area, Average Segment Length, Edge Directional Histogram Distances, and Number of Segments. A segment in this context is simply any visible photoreceptor in a Bio-Image. These criteria are what we call "Extracted Features" because they characterize each image as a unique entity. If the user selects an image, depending on the criteria used, these features are compared to each other to yield the query results. The feature extraction programs, which are used to extract the search criteria from the images, are implemented in the Matlab programming language and environment. Once all of the features are acquired, they are written to an external text file, which is transferred to a Linux server where the search engine is hosted. I implemented PHP scripts and programs that do all of the background work for the search engine and placed them into four different categories: Database Population, Search Conducting, Main Interface, and User selection. Database Population scripts are small PHP programs that insert all of the extracted features into a MySQL database where they will be placed into their respective tables. Search Conducting programs are the search handlers and they are responsible for determining which of the Bio-Images and features should be returned when they are given a query. The Main Interface is the visual portion of the search engine. Finally, the User Selection codes are the modules that actually allow the user to choose which image they want to use as a query.

Multi-Criteria Search on Retinal Images

Joriz De Guzman, CSUSB
Laura Boucheron, BS Manjunath, ECE UCSB

Research Objectives

- Create a Search Engine for Feline Retinal Images
- Implement to support different search criteria: Edge Histogram Distances, Average Area, Average Length, and Number of Segments
- Return the Metadata corresponding to images in the query and results
- To provide Biologists with a tool to make Biological assumptions

Screenshots

Significant Results

- Implemented First Matlab script to extract the Edge Directional Histogram Distances.
- Created Matlab Script to extract the number of segments within an image.
- Created Matlab Script to extract the average Area of each segment.
- Created Matlab Script to extract the average Length of each segment.
- Implemented the PHP scripts to populate specific databases with the extracted features.
- Implemented the Search engine to search the Bio-Images using the different search criterion.

Implementation

System Architecture

Search Scheme

- General: All features will be extracted from separate and specific Matlab scripts.
- Extracted features will be written to an external text file which will be used to populate the database.
- Depending on the query type used, the PHP script will access the correct table and Retrieve the requested data and images

Database Architecture

University of California, Santa Barbara
Center for Bio-Image Informatics
2004 Summer Intern Program

WAREHOUSING AND INTEGRATION OF BIOLOGICAL DATABASES

Kevin Hawkins

University of California, Santa Barbara

Mentor: Vebjorn Ljosa

Faculty Advisor: Ambuj Singh

The amount of data available to biologists is growing exponentially. This data resides in many databases with differing information and semantics. Scientists need a way to use all relevant information from these databases to help determine what proteins or genes might be of interest. This emphasis on data-driven research creates the need to integrate the available databases to allow querying across multiple heterogeneous datasets. I have created a data warehouse composed of local copies of the protein-protein interaction databases MINT, DIP, and BIND. The warehouse has a global schema that allows users to formulate a query without having knowledge about the specific databases involved. A web interface provides an easy way to query the data warehouse. Results returned to the user may be a combination of information from different sources. The interface identifies to the user the source of each result, and when applicable, a confidence value for the result is also displayed.

In order to create a data warehouse containing the information from MINT, DIP, and BIND, I first made local copies of each in a relational database. Next I investigated the differences between two integration techniques, global-as-view and local-as-view, and how they apply to biological data. With global-as-view the global schema is defined in terms of the sources. In other words you explicitly state where in the sources you find the information for global fields. Local-as-view is the opposite, the sources are defined in terms of the global schema, basically describing what information is present in the source but not describing how it relates to the global schema. Local-as-view requires inference to resolve a global query since it is not explicitly stated where to find global information within the sources. A global query to global-as-view is simple because the global schema has already been defined in terms of the sources and you know exactly where to find the global information. Adding sources to the data warehouse using global-as-view requires rewriting the query, which describes the global schema, and making it more complex. In comparison, local-as-view only requires writing one more relatively simple query to describe what the source contains. For this project I decided to use global-as-view to learn first hand some of the advantages and disadvantages when integrating biological data. Future research using local-as-view would be interesting for comparison.

To integrate the proteins from multiple sources, I used some protein database identifiers that are already present. The two ID's I used are SwissProt and Genbank. DIP has SwissProt and Genbank identifiers associated with its proteins, MINT has SwissProt identifiers present and BIND has Genbank identifiers associated with its proteins. Using these identifiers I found as much overlap between the proteins as possible and put them all into a global protein table with no repeats, a unique global identifier, and fields to keep track of which databases they came from. Then for the protein-protein interactions I looked up the global ID for the proteins involved and used this information to discover where the interactions

overlapped. I then stored the integrated interactions in a global table with a unique identifier and fields to keep track of which databases the interaction came from. Information about the interaction including how an experiment was conducted, a confidence value in the interaction, and a PubMedID to the paper that the interaction is published in was is stored in a global table of evidence. This type of information proved difficult to integrate because experiments have slight differences from each other. For example, some variables changed slightly between experiments and there were experiments missing information. As a result of these problems the evidence was not integrated and instead all the evidence available for an interaction was stored in multiple entries.

A web interface has been designed to allow biologists to access the integrated data. The interface allows the user to search for a particular protein using many available protein identifiers and/or taxonomy and allows constraints to be put on the search according to which datasets the results are to come from. Then all matching proteins are presented to the user and a protein can be selected. All the information about the selected protein is displayed along with all the proteins that it interacts with. The interactions can be selected to display the evidence for the interaction or a specific protein can be selected to view a list of proteins it interacts with.

The final result is a data warehouse containing 40,539 proteins and 81,326 interactions which has been created by integrating the data from protein-protein interaction databases MINT, DIP, and BIND. Through my research I have been able to conclude three main points: 1) the integration of data from biological experiments is difficult because of slight variations in experiments and missing information between datasets, 2) current database systems are not efficient for biological data integration because of the string indexes used and complicated queries do not create index structures on the sub-queries, and 3) even when using three similar sources the queries to define the global schema using global-as-view are quite complex while answering global queries is quite simple.

Warehousing and Integration of Biological Data

Kevin Hawkins, UCSB
Vebjorn Ljosa, Ambuj Singh, Computer Science UCSB

Objectives

- Create data warehouse of protein-protein interactions from multiple heterogeneous datasets
- Investigate different data integration techniques including global-as-view vs. local-as-view and materialized vs. virtual
- Allow biologists to easily formulate queries and display relevant results

Approach

Start with protein-protein interaction databases MINT, DIP, BIND (available online)

Download datasets and create local copies

Integrate the datasets so that they fit into a global schema

Create a web based interface to query the database

Integration

- The proteins from each dataset contain at least one ID from another database (SwissProt, Genbank or both)
- Use these ID's to find where the 3 datasets overlap
- Combine into one global table of proteins

GlobalID	mintID	dipID	bindID	SwpID	GIID
1	NULL	G_31	NULL	P15923	105865
2	12004	G_12	NULL	Q09472	627657
3	11953	G_7	2654	P04637	8400738

Global Schema

Integration Techniques

Global-as-view

Global schema defined in terms of source views

Pros: Easier to reformulate global query in terms of source schema, a matter of unfolding

Cons: When adding a source a complex transformation must be def

VS.

Local-as-view

Sources defined in terms of the global schema

Pros: Easier to add a source: just describe what is in the source in terms of the global schema

Cons: Global query requires inference to figure out reformulation

Results

A materialized data warehouse with global-as-view, containing 40,539 proteins and 81,326 interactions.

Global Proteins

Total = 40,539 Proteins

Global Interactions

Total = 81,326 Interactions

Conclusions

- Integration of data from experiments is difficult because of slight variations in experiments and missing information between datasets
- Current database systems are not efficient for biological data integration because joins using string indexes are inefficient and complicated queries do not create indexes within sub queries
- Even with 3 similar sources the transformations for global-as-view are complex

Acknowledgements

This research was supported in part by the National Science Foundation under grant EIA-0080134 and by the National Science Foundation Integrative Graduate Education and Research Traineeship (IGERT) program under grant no. 022713.

"I never knew much about graduate school before this program. Everyone in the lab shared their experience and helped me understand the benefits and opportunities grad school has to offer."

IMAGES OF RETINAL DETACHMENT ON A MOUSE MODEL

Lorraine Dansie

California State University, San Bernardino

Mentor: Mark Verardo

Faculty Advisor: Steven Fisher

One of the main goals of the research in this summer was to take the initial steps toward building a searchable database of cellular and sub-cellular microscopy images. This involved collaborative interaction between the biologists generating and analyzing the images and the engineers warehousing the images and developing data mining paradigms. One aspect of the study was to generate images and record the pertinent information that would be entered as the metadata for the engineers. Working with the Neuroscience Research Institute (NRI), I assisted in creating stained sections of normal and detached retina for a mouse model.

The retina is a thin, neural layer in the eye that is responsible for processing light into a visible image. Within the retina are several layers and within each layer there are sets of specialized cells that have unique functions in the processing of the visual information. In order to study the retina and its layers, eyes are embedded in agarose, a gel like substance, and cut into 100μm sections on a vibratome. Within each section of agarose is a sliver of retina that allows us to see the full spread of the layers. These layers are then stained with a two antibody, immunohistochemical, staining process that allows viewing of specific cell types within the retinal layers.

First, a primary antibody is applied that recognizes a specific protein. Then a secondary antibody, which is conjugated to a fluorophore, is added that recognizes the primary antibody. Several of the secondary antibodies can bind to one of the primary antibodies creating a higher visibility, due to the fluorophores. The cell(s) that contains the protein of interest can then be viewed under the microscope. This staining process is done on normal and detached retinal sections so comparisons can be made to help increase understanding of the visual process. For this study, detachment occurs when the retina separates from its top layer: the photoreceptors separate from the pigment epithelium layer. The photoreceptors consist of the rods and cones. Rods allow the distinction of shape, movement, light and dark while cones give the ability to distinguish different wavelengths of light, or "see in color." It is within the rods and cones that the visual processing begins. When this layer separates from the top layer, many changes occur throughout the retina. Some cells hypertrophy, or grow and extend into other retinal layers. Some cells "flatten," as in their shape actually changes, and others change expression of proteins. With all these changes occurring, it is of great interest to understand why they happen and how they affect vision when the retina is detached.

My work this summer consisted of doing the staining runs on mouse retinal sections, scanning the images into the computer, recording the metadata and interacting with the engineers once this

information was obtained. I began by reading about the retina, researching antibodies and, then searching for an antibody that would successfully stain Müller cells and astrocytes which are two types of glial cells. In my research I found a different Vimentin antibody that did just that. The antibody was ordered and I performed the first run. It was successful and now allows for new images to be created for better study of Müller cells and astrocytes. Glial cells form architectural support structures across the retina and regulate the ionic and chemical environment. Müller cells and astrocytes are two cell types that about which little is known. This new antibody will allow the lab to research these cells further, in hopes of building new understanding for their roles in normal and detached retina.

To further the interaction and understanding between the sciences, bi-weekly presentations were given by each of the interns to explain their summer work. We each gained a better understanding of the biology behind the images and the processes that go into adding images, creating the schema for the images, and making them searchable within designated parameters. To complete the summer, I created a poster outlining this work and gave a final power-point presentation to all faculty, staff and graduate students involved in this project, detailing our work and our progress.

Images of Retinal Detachment in a Mouse Model

Lorraine Dansie*, Mark Verardo, Steven Fisher, Neuroscience Research Institute, University of California, Santa Barbara, *California State University, San Bernardino

INTRODUCTION:

What is the Retina?

- Thin neural layer in the eye
- Processes visual information
- Retinal detachment is a common form of sight-threatening injury in humans

What animals are studied?

- Mouse
- Cat (control)

What antibodies are used?

- Vimentin [1:500]
- GFAP [1:800]

METHODS AND MATERIALS:

What is Immunohistochemistry?

- Staining process
- Two antibodies
  - Primary (1°AB)
  - Secondary (2°AB)
- 1°AB recognizes specific protein sequence
- 2°AB recognizes 1°AB
- 2°AB is conjugated to a fluorophore

DISCUSSION:

What cells are studied?

- Glia:
  - Two types:
    - Muller Cells
    - Astrocytes
  - Form architectural support structures across the retina
  - Regulate the ionic and chemical environment

CONCLUSION:

Anti-Vimentin

Anti-GFAP

- These antibodies will enable us to study the role of glial cells in retinal detachment in a mouse model
- These images will be added to the searchable Bio-Imaging database

University of California, Santa Barbara
Center for Bio-Image Informatics

2004 Summer Intern Program

CHARACTERIZING HIGLY ORDERED PYROLYTIC GRAPHITE USING ATOMIC FORCE MICROSCOPY

Michael Conry

California Institute of Technology

Mentor: Brian Piorek

Faculty Advisor: Sanjoy Bannerjee

Highly ordered pyrolytic graphite (HOPG) was studied using atomic force microscopy (AFM) techniques. The atomic force microscope was driven in contact mode with a CSC-21 tip. Contact mode is achieved by placing a cantilever so close to the surface of the substrate that it is in the repulsive region of the Leonard Jones potential. The tip is held along the surface using a constant force. As the tip moves with the contours of the surface, the cantilever and tip bend with the surface. A feedback loop then adjusts the vertical position of the cantilever to return it to the constant force. In this way, we can track the location of the head as a topographical map of the surface. The first goal in this summer program was to get a basic understanding of the physics behind the AFM and become able to steer the equipment to take atomic resolution images.

The high resolution images would ultimately be of HOPG. HOPG is a carbon configuration in which there is a stable lattice along the XY basal plane; however between layers there are only van der Waals forces. This causes the HOPG to be layered in sheets that can be removed using scotch tape (similar to Mica). This property was used to create fresh atomically smooth surface for each experiment trial. After cleaving, the HOPG was oxidized in oxygen at atmospheric pressure and 625° C. This creates monolayer and multilayer holes at the defect sites in the lattice.

After oxidation, we used the AFM to locate and study these holes. We would first scan a broad area of up to 20x20 microns to locate some monolayer circular holes. These holes would first appear in the lateral force scans at this resolution. Once located, we zoomed in on the hole to an area of .5x.5 microns. We observed monolayer holes that were .37nm deep and up to 20nm in diameter. Using the proper control parameters we were able to resolve the edge structure of the holes. Using Matlab to average the scan cross-sections across the middle, we can obtain a good image of the edge structure.

In future work, we hope to use these scans and the slope along the edge to understand the electric field created between the layers in this area. This could be the cause of certain edge artifacts that appear when scanning the holes using other microscopy and AFM scan modes.

Highly Ordered Pyrolytic Graphite Studied with Atomic Force Microscopy

Michael Conry, California Institute of Technology

Brian Piorek, Sanjoy Banerjee, Chemical Engineering, UCSB

Research Objectives

- Obtain high resolution AFM images of HOPG
- Characterize the edge structure of both monolayer and multilayer pits

Approach

1. Cleave HOPG sample with scotch tape.
2. Oxidize sample in furnace at 620° C for 25 minutes
3. Scan with PSIA XE-100 atomic force microscope using CSC-21 tip from MikroMash
   a. Observe course structure to determine quality of the oxidation
   b. Zoom in to be able to resolve multilayer pits topographically and monolayer pits through lateral force perturbations
   c. Zoom in further to observe fine structure of monolayer pits
4. Analyze data in XEI and MatLab to observe edge structure

AFM?

Cantilever moves touches the surface

Cantilever tip bends, creating greater force on the head

Feedback "corrects" the force by moving the head, tracing the topography of the surface

Results and Data

Circular Holes

.38 nm deep, 40 nm wide

.37 nm deep, 85 nm wide

Hexagonal Holes

7.5 nm deep, 200 nm wide

2.5 nm deep, 150 nm wide

Impact

- Better understand a commonly used substrate for AFM bio-imaging
- Localizing of imaging biological interactions giving better control

University of California, Santa Barbara
Center for Bio-Image Informatics

2004 Summer Intern Program

"This program opened up the opportunity of graduate school to me, allowed me to gain a lot of information, and solidified my decision to attend graduate school."

EFFECTS OF DIFFERENT TAU ISOFORMS ON KINESIN-DRIVEN MOTILITY: IMPLICATIONS FOR AXONAL TRANSPORT AND NEURODEGENERATIVE DISEASE

Riki Stevenson

California State University, Channel Islands

Mentor: Austin Peck

Faculty Advisor: Stu Feinstein

BACKGROUND

Microtubules (MTs) are a cytoskeleton component which confer shape to cells and allow for intracellular transport of vesicles and organelles. They can serve as conveyer belts within a cell distributing materials. Function and structure are highly correlated and in many cases dependent on various microtubule-associated proteins (MAPs). Microtubules can be extremely dynamic, as needed to capture and pull chromosomes apart in a dividing cell or to help cells move. They can also be very stable, as seen in adult neurons whose position can be fixed for the life of the organism. Microtubule structure starts with a tubulin dimer comprised of alpha and beta subunits. These dimers join together in a head-to-tail formation creating structural polarity and protofilaments, which join together laterally to create a hollow cylinder about 25nm in diameter. While the structure is fairly simple, it allows for a considerable variety of functions. In neurons, axonal microtubules are arranged in parallel bundles with their minus ends toward the cell body and more dynamic plus ends toward the synaptic terminal. Regulation of MT dynamics is critical for proper function.

Tau is a neuronal microtubule-associated protein which binds to and helps stabilize microtubules. They are necessary for axonal pathfinding, growth, transport and morphology. There are six isoforms of tau found in the human brain which arise from alternative mRNA splicing of one gene. The two main classes of tau, called 4-repeat and 3-repeat tau (4R and 3R) are characterized by the presence of four or three repeats in the carboxyl terminal. Within these two groups the proteins are distinguished by the presence of zero, one or two deletions in the amino terminus termed long, medium, and short, respectively. For the purposes of this paper, the main area of study is the differences between 4R and 3R tau.

4R and 3R tau differentially affect dynamics and the ratio of the two (~1:1 in adults) is finely regulated developmentally. A change in the 4R:3R tau ratio towards 4R can cause neuronal cell death. Many neurodegenerative diseases, such as Alzheimer's, exhibit abnormal pathological fibers composed mainly of tau, called neurofibrillary tangles (NFTs). Tau misregulation has also been correlated to neuronal cell death and dementia in frontal-temporal dementia and parkinsonism associated with chromosome 17 (FTDP-17).

Kinesin is a microtubule-associated motor protein comprised of four proteins, two heavy chains and two light chains. The heavy chains form the two heads which walk along MTs like feet with an 8nm step. The two light chains form the tail and attach through other proteins to vesicles, organelles or other materials that are

distributed throughout the cell, along MT tracks in a direction-specific manner. Diversity of kinesin within the KIF superfamily allows for plus-end and minus-end directed transport of materials throughout all reaches of the cell. For our purposes, we are referring to KIF5, a neuronal, axon and plus-end directed kinesin. Recent studies suggest that axonal kinesins may recognize a special dynamic cue in the MTs of the initial segment of the axon.

The tau misregulation could bring about a loss-of-function which then triggers programmed cell death. The misregulation of tau isoforms could impair axonal transport via three possible mechanisms, which are not mutually exclusive. These include: 1) a tau-induced alteration of the MT dynamics needed to orient kinesin, 2) steric hindrance of kinesin motility due to irregular tau-tau binding and 3) a tau-induced MT-conformation that is unfavorable to kinesin motility.

Important to this assay is also establishing a consistent, reproducible source of KIF5 which will be used to further this experiment and imaging data.

PRELIMINARY DATA

Preliminary data suggest that taxol and tau isoforms demonstrate differing abilities to modulate kinesin motility. 4R affects kinesin motility differently than taxol and 3R with statistical significance. Since the ratio of 3R:4R is finely regulated throughout development, suggesting functional differences, any disturbance of the ratio could strongly affect transport of cellular components along axons.

METHODS

Through *in-vitro* assay, by Video-enhanced Differential Interference Contrast (DIC) microscopy, we examine the effects of 3-repeat tau (3R) versus 4-repeat tau (4R) on motility. In DIC microscopy, the shadow of the image is observed instead of the direct image itself. A flow-chamber schema is used to create slides for visualization. Two pieces of double stick tape and a cover slip create the flow chamber. Kinesin is pulled through the chamber by capillary action and the tails attach themselves to the glass slide through a difference in charges. Tau-stabilized tubes are pulled through and attach to the heads of the kinesin. Movement of the MTs is visualized as the kinesin pass the MTs in a hand-over-hand motion above the heads of the protein. The process of viewing the MTs moving is much easier to visualize since they are much larger. Data is recorded on DVD and rates of MT movement are computer analyzed.

Recombinant DNA was utilized to create pure KIF5 protein. pET-17b K560GFP was inserted into (transformation) and multiplied by XL1Blue bacteria and purified by QIA Prep Kit. An agarose DNA gel was run after restriction digest to verify. Protein expression was obtained in BL21(DE3) bacteria after transformation and by induction of the lac operon with isopropylthiogalactoside (IPTG), an artificial inducer of the lac operon. A Coomassie-stained SDS PAGE was run of uninduced vs. induced bacteria. Protein seemed to be expressed in the induced bacteria where the kinesin was predicted to run (~100kd). For more certainty, a western blot was performed using Rabbit anti-GFP and Donkey anti-rabbit. The western blot showed the presence of the kinesin-GFP, but also showed the presence of a leaky promoter in the vector DNA. Kinesin was purified from E.coli lysate by nickel-column with His-tag affinity.

DISCUSSION

If the rate of kinesin-motility is affected differentially by tau isoforms, altered axonal transport could be a molecular-level cause of several neurodegenerative diseases. The recombinant kinesin produced will not only be used to further this particular research, but also to create additional images to be utilized in a searchable Bio-molecular image database.

EFFECTS OF DIFFERENT TAU ISOFORMS ON KINESIN-DRIVEN MOTILITY: IMPLICATIONS FOR AXONAL TRANSPORT AND NEURODEGENERATIVE DISEASE

Riki Stevenson*, Austin Peck, Michelle Massie, Stuart C. Feinstein and Leslie Wilson, Neuroscience Research Institute and Department of Molecular, Cellular, and Developmental Biology, UCSB, CA * Cellular and Molecular Biology, CSUCI

Background:

➢**Microtubule** MT)

➢A cytoskeleton component which confers shape to cells and allows for intracellular transport of vesicles and organelles

➢Function largely depends on associated proteins

➢**Tau**

➢A neuronal MT-stabilizing protein

➢6 isoforms derive from alternative mRNA splicing and differentially affect MT dynamics.

➢Contributes to axonal growth, transport, polarity and morphology.

➢**Kinesin**

➢Vesicle-bearing, processively-walking motor protein

What Is Known:

➢A change in the ratio of 3-repeat:4-repeat tau (3R:4R) (normally ~1:1 in adults) towards 4R tau can cause neuronal cell death

What Is Thought:

➢Tau misregulation, in the form of a 3R:4R tau ratio shift, could cause a loss-of-function in neurons

➢Axonal transport could be the function neurons lose when they stimulate programmed cell death

➢"Window" of optimal function

Three possible mechanisms of Tau-Mediated Neurodegeneration

Methods:

➢Create a consistent, limitless source of neuronal kinesin via cDNA vector

➢Transformations, protein expression, purification

➢Flow chamber analysis of MT movement due to kinesin-motility while varying tau isoforms and taxol

Results:

➢pET-17b K560GFP vector

➢DNA Gel after 1st transformation and restriction digest

➢Hind III & Pvu II

➢Coomassie-stained 7% SDS PAGE of plasmid expressing cells

➢Western Blot

➢1° antibody Rabbit Anti-GFP, 2° Donkey Anti-rabbit

Discussion/Impact:

➢Tau misregulation has been correlated to neuronal cell death and dementia in frontotemporal dementia and parkinsonism associated with chromosome 17(FTDP-17). Many neurodegenerative diseases, like Alzheimer's, exhibit abnormal pathological fibers composed mainly of tau.

➢If the rate of kinesin-motility is affected by a change in tau ratio, altered axonal transport could be a molecular-level cause of several neurodegenerative diseases.

➢Example of images to be used in a Bio-Image database

growing vs. shortening

0sec 7sec 30sec

University of California, Santa Barbara
Center for Bio-Image Informatics
2004 Summer Intern Program

"I liked the academic environment. The use of diverse tools being combined into one application was something I found interesting as well."

A DATA STORAGE, PROCESSING, AND RETRIEVAL SYSTEM FOR MICROTUBULE TRACKING DATA

Robert Coulier

California State University, Channel Islands

Mentor: Arnab Bhattacharya

Faculty Advisor: Ambuj Singh

Microtubules are cytoskeleton components of cells. Given a set of cell microtubule images taken at regular time intervals, individual microtubule positions can be tracked over time. The process can be subdivided into two portions. The first part involves position data acquisition from microtubule images and videos. The second portion involves the processing of this raw data into meaningful statistical results. The scope of this project was to automate the second portion of the experimental process using the tools of computer science. This was divided into 3 phases. Since this tracking process is one of the central elements of cell microtubule research, it is hoped that this project will facilitate research in this area.

Position data can be analyzed to obtain event information for an individual microtubule as well as statistical data for a group of microtubules. An event can be one of three primitive types: growth, shortening, or attenuation. These events describe microtubule lengthening, shortening, and relative inactivity. In addition, event catastrophes indicate periods of microtubule growth followed by periods of microtubule shortening. Rescues indicate periods of microtubule shortening followed by periods of microtubule growth.

Phase 1 of the project involved automated event identification. To accomplish this, position data first had to be extracted from data files kept in Microsoft Excel format. This was implemented through the java programming language utilizing a third party library to handle the excel data extraction. Coordinate information in the form of x and y plots were then converted into length displacements. This data was further processed to obtain slope information that varies with time. By comparing any given slope with a predefined attenuation threshold an event could be determined depending on whether the slope exceeded, fell within the range of, or dropped below the attenuation threshold. After event data was obtained, statistical analysis was done to obtain totals, averages, standard deviations, etc. of each event type. An additional statistic called dynamicity was also calculated.

Phase 2 of the project required the storage of raw and derived data into a relational database. Major work in this phase involved the development of a database schema capable of adequately representing the data, which is of interest to the microtubule researcher. The resultant schema included a Position table used to represent raw tracking data, an Event table used to represent event information, a Stack table used to represent identifying characteristics of a microtubule group, and a Parameters table containing all derived data and related statistical information. The database was implemented using MySQL.

Phase 3 called for the development of a web accessible user interface. The query form had to be flexible enough to allow for a broad range of possible queries. The interface that came to fruition was a sum of products query structure which allowed the user to enter a variable number of constraints contained within a variable number of constraint sets. The query form had to be dynamic requiring the use of the JavaScript client side programming language.

The project phases were incorporated into what is a MVC (Model View Controller) software architecture. This design is modular and scalable to the future needs of microtubule researchers. The system now provides data processing support and capability, which is to be incorporated into a larger bioimage bioinformatics project: a searchable biological image database.

Microtubule Tracking Database

CSUSB

Robert Coulier, CSUSB

Arnab Bhattacharya and Ambuj Singh, Computer Science - UCSB

Research Goals

- Automate the processing of microtubule tracking data
- Store raw and derived data in a relational database
- Create a visual interface for data retrieval

Methods: Data collection

Methods: Event tracking

Design: Database schema

Design: User interface

Features

- Highly dynamic
- Sum of products query structure

Impact: The larger project

The tracking database will provide data processing support for a larger bioimage searchable database.

"[The Bio-Image Program] has shown me that there are good opportunities in pursuing further education."

CREATING A WEB ACCESSIBLE INTERFACE TO AN IMAGE DATABASE

Timothy Berger
Computer Science
Cal State University, San Bernardino
Mentor: Zhiqiang Bi
Faculty Advisor: B.S. Manjunath

With advances in current technology, biologists have the ability to image various biological samples at an increasing rate. However, the biologists lack a tool in which to perform analysis and comparisons on these increasingly growing image collections. The project, which I have implemented, is a tool in which the biologists can view large collections of images over the web (for easy accessibility), as well as perform similarity searches on image collection.

The first step in implementing this tool was to design architecture for the system in such a way as to facilitate maintenance and expandability. These factors were key to implementing a fully functional application. With this architecture in place, the process of implementation can begin. To implement the user interface I used a combination of PHP and JavaScript. PHP is a server-side scripting language that facilitates the development of data driven web applications. JavaScript is used to receive client-side requests and to provide a robust interface for the user. C++ was used for the actual core components of the system, such as the searching and feature extraction applications. The decision to use C++ to implement the core components was due to the speed advantage and functionality of this language. Then extracted features were placed into the MySQL database for later use.

With the basic framework in place the task implementing to core system began. This core was a collection of routines that were used to extract various features from the images, such as pixel average, pixel variance, histogram values, and homogenous texture descriptors. These routines were used to perform searches on these extracted features. The method that returned the best results, in terms of similarity, was searching over the homogeneous texture descriptors. The homogeneous texture descriptors are a collection of 64 numeric values that define an images texture uniquely.

The process used for searching over the images was to take the Euclidian distance of the feature set from the user's selection and the feature sets in the database. The Euclidian distance formula is used to measure the distance between two sets of data. With the numeric value returned from the Euclidian distance equation the images could then be ranked based on similarity.

Performing a linear search on these collections of images can become quite inefficient as these collections grow. A method to solve this problem was to implement a clustering algorithm. Clustering essentially groups all similar elements in a collection. The process I used to cluster my data sets was the K-Means clustering algorithm. The steps of this process are as follows: 1) Place K random points into your data set, these represent your cluster center points, 2) with these centroids in place calculate the distance from each element in the set to each centroid point and group the element with the closest centroid point, 3) take the average of all the values in the cluster, this will be the new value of the centroid point for that cluster. Repeats steps 2 and 3 until the centroid points cease to shift. The advantage of clustering is that instead of searching over the entire collection for similar images, the user will only need to search over the closest cluster to get similar images, which reduces the search set substantially.

With the completion of this tool, the biologist will now be able to keep up with increasing speed of technology and automate many repetitive and time consuming tasks.

WEB INTERFACE FOR A SEARCHABLE IMAGE DATABASE

Timothy Berger, CSUSB

Zhigiang Bi, B.S. Manjunath, Ambuj Singh, UCSB

Project Objectives

- Create a user interface capable of displaying images to the user
- Create a database schema used for storing the various data sets retrieved from an image
- Implement various data mining application to find relevant information in an image
- Use various searching techniques to return similar images based on a user's query

Interface Layout

Thumbnails

Enlarged Image

Database Schema

- Relational Structure
- Reduce redundancy

Data Extraction

- Extract simple statistics, i.e. pixel average, pixel variance, and histogram values
- Extract Homogeneous texture descriptors. 64 values that uniquely define an image's texture.
- Homogeneous textures extracted by entire image and sub-images.

Search Method

- Gather data for the selected images
- Calculate the distance from the selected image's dataset to each corresponding dataset in the database.
- Euclidean distance formula used to calculate the distance between the images.

$$\sqrt{\sum_{i=0}^{n} (a_i - b_i)^2}$$

Clustering

- Sequential search slow over large datasets
- Clustering improves search speed
- K-Means clustering utilized

Step 1

Choose K number of points in your dataset. These represent your centroid points

Step 2

Calculate the distance from each element into the collect to each centroid point. Group with the closest centroid point

Step 3

Adjust each centroid point to be the average value of the elements in the cluster then repeat from Step 2

University of California, Santa Barbara
Center for Bio-Image Informatics

2004 Summer Intern Program

"Seeing the whole research element of graduate school added to my interest."

AUTOMATED FEATURE EXTRACTION FROM RETINAL IMAGES

Victor Arellano

Computer Science

Cal State University, San Bernardino

Mentor: Samuel Frame

Faculty Advisor: Sreenivasa Rao Jammalamdaka

With the large and constant flow of retinal images that are produced it has become quite a significant task to analyze each image. This is something that is normally done by hand and can be quite laborious because of the sheer amount of images. The goal here was to develop a program that can take each image and process it in a way that all the necessary data such as means, variances, etc can be extracted automatically. The program would then be able to output all the necessary data to anything that may require it such as a database.

With these overall goals in mind, a programming module was written in the Java language utilizing an open source library called Java Imaging Utilities (which can be found at the address: http://jiu.sourceforge.net). This library provides the groundwork for the program. It enables the loading of the retinal images in an interpretable form that the computer can operate on.

The program starts off its analysis routines by first gathering the image data into a two dimensional array before attempting calculations. It then moves on to normalize the image, which enable it to perform further calculations and essentially create a bivariate probability mass function. The first of the calculations that it performs is the mean calculation. Means are simply a measure of where the image data for a particular channel is concentrated. An orientation check is then performed utilizing a method involving these means and then the array is rotated to correct the orientation if necessary. It is necessary to perform this task so that the results can be standardized across all images. Covariances and an eigenvalue ratio are the last things that the program currently calculates.

By automating the process of analyzing these images it enables work to be accomplished much faster and allows people to move their attention elsewhere while the program does its job. This also allows anyone to perform the feature extraction from the images. They will not be required to have a mastery of how to do the calculations necessary for the feature extraction itself, but instead can simply run the program on the desired images and have the task performed for them. Also, the program is written in a manner so that it can easily be expanded upon. It can be further optimized and outfitted with more feature extraction functions.

Automated Retinal Image Feature Extraction

Victor Arellano, CSUSB
Samuel Frame, Sreenivasa Rao Jammalamadaka, Statistics - UCSB

Project Objectives

- Develop a coded module to extract various features from retinal images that exhibits the following features:
  - Executes in a reasonable amount of time per image
  - Runs & produces results in a semi-unsupervised fashion
  - Can be set to analyze many images at once
- It will be beneficial to maintain portability with the program so that it can be transferred to other languages or easily used within another program

Impact

- The point of the program is to make simple feature extraction a fast and semi-unsupervised process
- Since the program can figure out data that is needed for each image it allows people to concentrate their resources elsewhere
- The data can then be used elsewhere for example in a database or other program
- Expansion is possible on the program to make it calculate anything else that may be needed in the future

Program Flow

Implementation

- The program is coded in Java and utilizes the open source Java Imaging Utilities library for some basic imaging function
  - http://jiu.sourceforge.net/
- Organized to take in one image after the other and run until all images have been processed
- Loads & orients images, gets dimensions, calculates means, covariances, & eigenvalues

Examples of Extracted Data

Orientation Correction

Mean Calculation

Covariance Calculation