C-DEM: A Multi-Modal Query System for Drosophila Embryo Databases

Fan Guo, Lei Li, Christos Faloutsos, Eric P. Xing
School of Computer Science
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213, United States
{fanguo, leili, christos, epxing}@cs.cmu.edu

ABSTRACT

The amount of biological data publicly available has experienced an exponential growth as the technology advances. Online databases are now playing an important role as information repositories as well as easily accessible platforms for researchers to communicate and contribute. Recent research projects in image bioinformatics produce a number of databases of images, which visualize the spatial expression pattern of a gene (eg. "fj"), and most of which also have one or several annotation keywords (eg., "embryonic hindgut").

C-DEM is an online system for Drosophila (= fruit-fly) Embryo images Mining. It supports queries from all three modalities to all three, namely, (a) genes, (b) images of gene expression, and (c) annotation keywords of the images. Thus, it can find images that are similar to a given image, and/or related to the desirable annotation keywords, and/or related to specific genes. Typical queries are *what are most suitable keywords to assign to image insitu28465.jpg* or *find images that are related to gene "fj", and to the keyword "embryonic hindgut"*. C-DEM uses state-of-the-art feature extraction methods for images (wavelets and principal component analysis). It envisions the whole database as a tri-partite graph (one type for each modality), and it uses fast and flexible proximity measures, namely, random walk with restarts (RWR).

In addition to flexible querying, C-DEM allows for *navigation*: the user can click on the results of an earlier query (image thumbnails and/or keywords and/or genes), and the system will report the most related images (and keywords, and genes). The demo is on a real Drosophila Embryo database, with 10,204 images, 2,969 distinct genes, and 113 annotation keywords. The query response time is below one second on a commodity desktop.

1. MOTIVATION AND SIGNIFICANCE

The goal of the C-DEM system is to provide a flexible way of querying and navigating large databases with biological images, and specifically annotated Drosophila (= fruit-fly) embryo images. The problem has several challenges: (a) the ideal system has to be multi-modal, because we want multiple data types (images, text annotations); (b) it has to be flexible, since domain experts do not always have a specific query to ask; instead, they try to develop and test hypotheses, where the ultimate, 50-year goal is to find a regulation map of thousands of genes in a Drosophila genome, and study how it evolves during the development of embryos in the first few hours or longer; (c) the system has to be fast, to handle the growing volume of experimental data; (d) the system has to be user-friendly, ideally requiring a few mouse-clicks per query, as opposed to an elaborate query language.

The significance and impact of such a system is growing. High-throughput technologies has become quite popular in bioinformatics research, providing a low-cost solution to data generation. Such data explosion makes biological databases more and more important in scientific discovery and communication. The number of molecular biology databases grows from less than one hundred in 1998 to almost one thousand by the end of 2007 [10], where the size of GenBank [7], one of the most accessed database of biological sequences, doubles approximately every 18 months [3]. The accumulation of experimental evidence would grant ever stronger power of databases in scientific discovery, especially with the help of statistical methods developed in data mining and machine learning.

Our system is based on one of the few most accessed datasets of biological images. It is released by the Berkeley Drosophila Genome Project (BDGP), and consists of more than 70,000 digital photographs documenting the expression patterns of more than 3,000 genes during the development of Drosophila embryos, annotated with a standardized set of terms for developmental anatomy [1]. Users could query and browse the database by gene name or its synonyms (eg. CG10917 or "fj"), which display a summary page of corresponding image thumbnails and annotation keywords, as well as other relevant information of gene expression. However, more complicated queries like *what are most suitable keywords to assign to image insitu28465.jpg* or *find images that are related to gene "fj", and to the keyword "embryonic hindgut"* cannot be answered by BDGP and, to our knowledge, any other existing query system for biological databases of images and texts.

In this paper, we propose to demonstrate C-DEM, the CMU system for Drosophila Embryo Mining, which meets the challenges for fast, flexible and user-friendly querying of biological databases like BDGP. The software architecture of C-DEM de-associates the front-end web server and back-end computing engine with a clear and stable API. They are deployed on separate machines for better performance by distributing the workload, and it also makes future updates of either component of the system easier. Implementation of the system also includes offline data pre-processing, image feature extraction, and construction of a graph representation of the

VLDB '08, August 24-30, 2008, Auckland, New Zealand

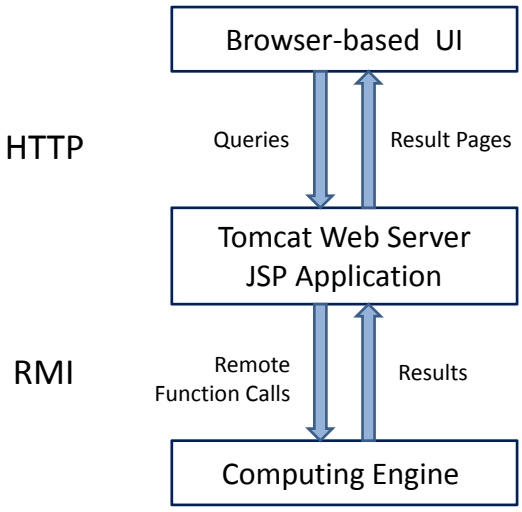


Figure 1: The software architecture of the C-DEM system. It consists three tiers: the browser-based UI, the Tomcat web server and the computing engine. They communicate via HTTP and RMI protocols.

dataset. The back-end computing engine loads the graph representation and estimate the proximity from query to a target of any type using random walk with restarts (RWR), for which fast algorithms already exist [17]. Algorithmic details of the system are presented in [12]. The C-DEM demo can be accessed by visiting http://www.db.cs.cmu.edu/db-site/Projects/cdem.

The remainder of the paper is organized as follows: Section 2 discusses the software architectures and describes the implementation of core computational part. Section 3 demonstrates query and navigation operations, with screen shots of the browser-based UI, for a simple example and a more complicated one. Related works are reviewed in Section 4, and Section 5 concludes this paper.

2. SYSTEM DESCRIPTION

As shown in Fig. 1, the system architecture of C-DEM consists of three tiers. A user interacts with the web server through the browser-based interface (the first tier), which sends the query information via the HTTP protocol. The second tier is hosted on an Apache Tomcat web server with JSP pages to accept and process user inputs. It issues queries to the computing engine (the third tier) by making remote function calls (Java Remote Method Invocation). The computing engine then performs the RWR algorithm on the pre-computed graph representation (discussed in Sec. 2.2) given the query inputs and a few pre-defined parameters. Results are first transferred back to the web sever using the RMI interface. Then HTML pages are generated and sent to the browser.

The remaining part of this section focuses on the implementation of the key component: the back-end computing engine, and we first discuss how to pre-process the image data and extract meaningful features.

2.1 Image Pre-processing and Feature Extraction

Our data collection comes from a subset of digital photographs in the BDGP database which are taken in the same development stage, such that the expression patterns of different genes are comparable. These raw images are of quite different image quality, due to focusing, noise, occlusion and other variable experimental

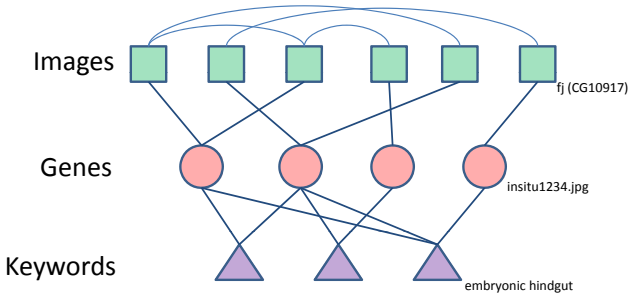


Figure 2: The tri-partite graph representing multi-modal data. Genes (eg. "fj" or "CG10917") are connected to both Images (bitmaps, eg. "insitu1234.jpg") and Keywords (ASCII strings, eg. "embryonic hindgut"). Images are also connected to their nearest neighbors in the 24-d feature space.

conditions. We pre-process them as described in [14], to do segmentation, de-noising, and to avoid overlaps and partial embryo images. A final registration step translates, scales and rotates embryos to a "canonical" orientation, and produces 8-bit gray level images of size 352×160. We choose a combination of local and global features derived from wavelet transformation and principal component analysis (PCA) respectively. After this feature extraction step, each image is represented by a 24-dimensional vector of real numbers between $0 \sim 1$. It is exactly these feature vectors that we use to measure the similarity between two images in the graph construction.

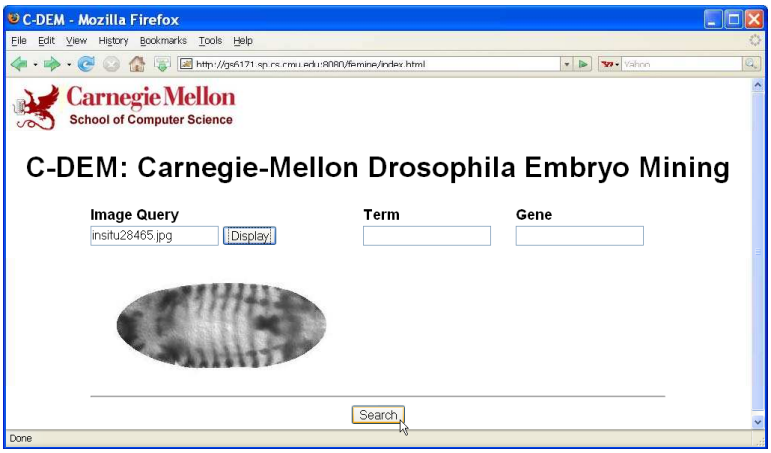
2.2 Graph Construction and Random Walk with Restart

We represent images and their attributes (the corresponding gene and annotation keywords) as nodes in a graph and link them together with undirected edges. Figure 2 is a schematic view of the structure of the graph. Each image node is connected to a gene node according to its attribute. Annotation keywords are connected to genes instead of images because in a fixed developmental stage, images of the same gene have identical annotations. Between image nodes, we compute Euclidean distances of image feature vectors obtained in the previous subsection, and connect the nearest neighbors together.

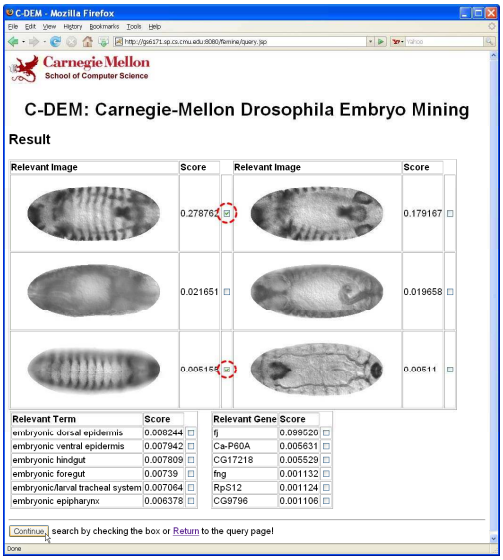
The next goal is to measure the proximity of two nodes on such a graph. This is exactly what the random walk with restarts (RWR), measuring the closeness of the querying node(s) to any target node. Given a connected graph and a query node set Q, the algorithm acts like a random walker who always starts his/her trip from a node randomly picked from Q. At every discrete time unit, the walker moves to one of the neighbors of the current node with probability $1 - c$ (random walk) or it returns to a random node in Q to restart the trip with probability c (restart). As time goes to infinite, the steady-state probabilities of this stochastic process is the "proximity vector" of every node to Q. The computation can be summarized in a single formula:

$$\mathbf{u} = (1 - c)\mathbf{A}\mathbf{u} + c\mathbf{v} \quad (1)$$

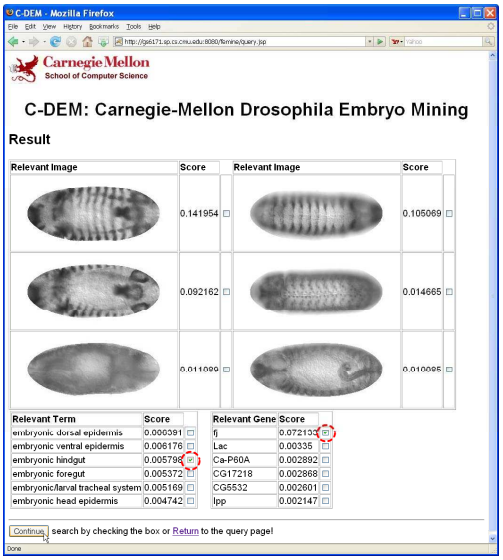
where $\mathbf{A}$ represents the adjacency matrix, c is the parameter of the restart probability, $\mathbf{u}$ represents the steady-state probability vector (desired "proximity" vector), and $\mathbf{v}$ represents the restart probability vector where all the elements are zero except that the element in Q. The algorithmic details are presented in [12].



(a)



(b)



(c)

Figure 3: (a) Index page with the preview of the querying image *insitu28465.jpg*. Search results in (b) include separate lists for images, genes and keywords (terms) ranked by relevance scores. By checking two images (circled with handcraft for emphasis) in (b) and clicking on the "Continue" button at the bottom, we can get to (c) which displays the new research results.

3. DEMONSTRATION

We first illustrate how to perform a straightforward query through the graphical user interface. Figure 3(a) shows the layout of the index page with an input image *insitu28465.jpg*. The user could preview the image by clicking the "Display" button to double-check, or directly go to "Search". The result page is shown Fig. 3(b) which listed outputs in all three modalities, with descending relevance scores, that is, steady-state probabilities from the RWR computation. The top left image is the original query; notice that similar patterns can be observed on other relevant images.

Now we can add another image from the search result into the query, and without returning to the index page, we can click the "Continue" button at the bottom left, which leads to the new results (Fig. 3(c)). Furthermore, C-DEM can carry out flexible queries from any modality; to illustrate that, we can issue a query from a gene and a keyword together (circled in (Fig. 3(c)). These two nodes are actually in the underlying graph representation, so the relevant image results in Fig. 4 could be interpreted as expression patterns of the gene *fj* which are most representative for the keyword *embryonic hindgut*. Then we can navigate by checking other keywords/genes/images, or we can follow the "Return" link to go back the index page.

4. RELATED WORK

In this section, we briefly review related works in three classes: multimedia query systems, database systems with ranking, and biological database systems.

There are a number of multimedia query systems like the early QBIC system [9] and the more recent MMG [15]. The first one allowed for similarity queries on images only (with no annotations or other attributes). The latter focused on automatic captioning and mining for a collection of images.

Examples of database systems with ranking are ObjectRank [6], BANKS [5] as well as the link-based ranking method in [11]. They all represent a database as a graph, and use either page-rank or spanning trees, to find important/relevant nodes, in response to

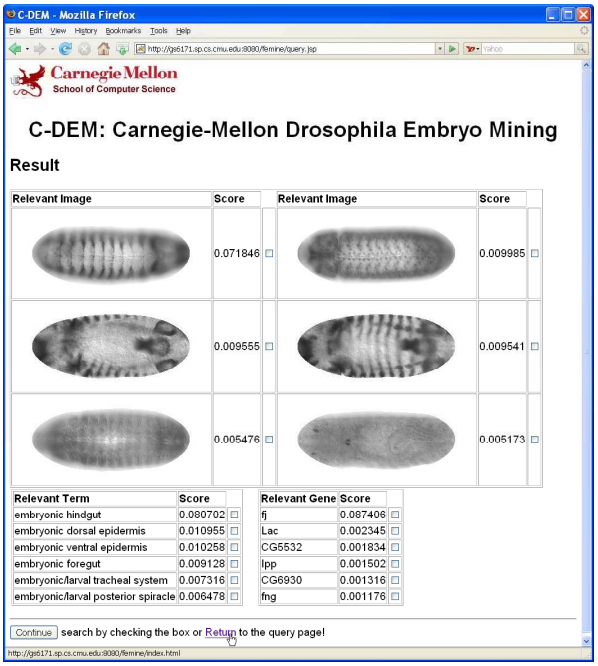


Figure 4: Result page for a cross-modal query from the gene *fj* and the keyword *embryonic hindgut* (selected in Fig. 3(c)).

the nodes that the user considers desirable. All of them are on semistructured data, with no obvious extensions to multimedia data types.

Online image databases of Drosophila include the Berkeley BDGP database [1], the Toronto Fly-FISH database [2] of fly embryos, the Oxford FlyTed database [4] of fly testes, etc. They only support search and browsing by gene name or synonyms. Systems like FEMine [14] focused on finding patterns and "eigenEmbryos" in a collection of images, using PCA or ICA. The FlyExpress [8] system which is based on BDGP data, provided search and navigation from image to image only.

In conclusion, C-DEM is the only system to our knowledge that combines multimedia, flexible querying, and ranking, for biological image databases.

5. CONCLUSIONS

We propose to demonstrate C-DEM, the CMU system for Drosophila Embryo Mining. C-DEM has the following desirable properties:

- It is multimodal, handling images, text and formatted attributes;
- It is flexible, allowing queries from any modality to any other(s);
- It is fast, with sub-second response time on commodity hardware and on a large real dataset;
- It is user friendly, allowing querying and navigation with a few mouse clicks.

Under the hood, we use state of the art image processing methods to extract features, and scalable algorithms to compute node proximities in a graph. Future work will include support for additional query functionality, like the "center piece sub-graphs" [16], and graph clustering using METIS [13] or spectral methods.

6. ACKNOWLEDGEMENT

This material is based upon work supported by the National Science Foundation under Grant No. DBI-0640543. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

We would also like to thank Polo Chau for his suggestions on the interface.

7. REFERENCES

[1] Berkeley Drosophila Genome Project, Patterns of gene expression in drosophila embryogenesis, Release 2, Mar 2007. http://www.fruitfly.org/cgi-bin/ex/insitu.pl.

[2] Fly-FISH: A database of drosophila embryo mRNA localization patterns. http://fly-fish.ccbr.utoronto.ca/.

[3] GenBank, NCBI-Genbank flat file Release 164.0, Feb 2008. Available at ftp://ftp.ncbi.nih.gov/genbank/gbrel.txt.

[4] Image Bioinformatics Research Group, Department of Zoology of University of Oxford, FlyTED: the drosophila testis gene expression database, Release 1.0. http://www.fly-ted.org/.

[5] B. Aditya, S. Chakrabarti, R. Desai, A. Hulgeri, H. Karambelkar, R. Nasre, Parag, and S. Sudarshan. User interaction in the banks system. In *ICDE*, pages 786–788, 2003.

[6] A. Balmin, V. Hristidis, and Y. Papakonstantinou. Objectrank: Authority-based keyword search in databases. In *VLDB*, pages 564–575, 2004.

[7] D. A. Benson, I. Karsch-Mizrachi, D. J. Lipman, J. Ostell, and D. L. Wheeler. Genbank. *Nucleic Acids Research*, 35:D21–D25, 2007.

[8] B. V. Emden, H. Ramos, S. Panchanathan, S. Newfeld, and S. Kumar. FlyExpress: An image-matching web-tool for finding genes with overlapping patterns of expression in drosophila embryos (2006), Arizona State University, Tempe, AZ. http://www.flyexpress.net/.

[9] C. Faloutsos, R. Barber, M. Flickner, J. Hafner, W. Niblack, D. Petkovic, and W. Equitz. Efficient and effective querying by image content. *J. of Intelligent Information Systems*, 3(3/4):231–262, July 1994.

[10] M. Y. Galperin. The molecular biology database collection: 2007 update. *Nucleic Acids Research*, 35:D3–D4, 2007.

[11] F. Geerts, H. Mannila, and E. Terzi. Relational link-based ranking. In *VLDB*, pages 552–563, 2004.

[12] F. Guo, L. Li, E. P. Xing, and C. Faloutsos. Mining fly embryo images using graph based methods. *SDM Workshop on Data Mining for Biomedical Informatics*, 2008.

[13] G. Karypis and V. Kumar. A software package for partitioning unstructured graphs, partitioning meshes, and computing fill-reducing orderings of sparse matrices - Version 4.0. Technical Report, Department of Computer Science, University of Minnesota, Minneapolis, MN, 1998.

[14] J.-Y. Pan, A. G. R. Balan, E. P. Xing, A. J. M. Traina, and C. Faloutsos. Automatic mining of fruit fly embryo images. In *KDD*, pages 693–698, 2006.

[15] J.-Y. Pan, H.-J. Yang, C. Faloutsos, and P. Duygulu. Automatic multimedia cross-modal correlation discovery. In *KDD*, pages 653–658, 2004.

[16] H. Tong and C. Faloutsos. Center-piece subgraphs: Problem definition and fast solutions. In *KDD*, pages 404–413, 2006.

[17] H. Tong, C. Faloutsos, and J.-Y. Pan. Fast random walk with restart and its applications. In *ICDM*, pages 613–622, 2006.