3D Medical Volume Reconstruction Using Web Services

Rob Kooper[*], Andrew Shirk[*], Sang-Chul Lee[*], Amy Lin[**], Robert Folberg[**] and Peter Bajcsy[*]

[*]National Center for Supercomputing Applications (NCSA)
University of Illinois at Urbana-Champaign (UIUC)
605 E. Springfield Avenue, M/C 476, Champaign, IL 61820
{kooper,shirk,sclee,pbajcsy}@ncsa.uiuc.edu
[**]Department of Pathology
University of Illinois at Chicago (UIC)
1819 W. Polk Street,446 CMW, Chicago, IL 60612
{alin, rfolberg}@uic.edu

Abstract

We address the problem of 3D medical volume reconstruction using web services. The use of proposed web services is motivated by the fact that the problem of 3D medical volume reconstruction requires significant computer resources and human expertise in medical and computer science areas. Web services were implemented as an additional layer to a dataflow framework called Data to Knowledge. In the collaboration between UIC and NCSA, pre-processed input images at NCSA were made accessible to medical collaborators for registration. Every time medical collaborators inspected images and selected corresponding features for registration, the web server at NCSA was contacted and the registration processing query was executed using the Image to Knowledge library of registration methods. Co-registered frames were returned for verification by medical collaborators in a new window. This paper presents 3D volume reconstruction problem requirements, architecture of the developed prototype system and the tradeoffs of our system design.

1. Introduction

Although web services attempt to solve many of the same problems as more mature, previous generation middleware technologies (CORBA, DCOM/COM+) they offer compelling advantages to consider when deciding how a distributed software application should be implemented.

Real system interoperability is one of the key advantages of web services. This interoperability, made possible by XML based standards, such as the Simple Object Access Protocol (SOAP) and the Web Services Description Language (WSDL). SOAP and WSDL are simultaneously compatible with and independent of any particular programming language. Many applications can tremendously benefit from using web services because there is a need to execute multiple processing algorithms on different operating systems and devices (potentially using Grid computing). Furthermore, these algorithms might be written using different programming languages and have originated from multiple vendors. Publicly available web service development frameworks now allow these objectives to be achieved at a very low cost.

Another major advantage of Web services is their orientation toward loosely coupled architectures. Since web service clients (end user applications and other web services) can dynamically bind with web services, complex orchestration can occur at runtime. Thus, investigating real-life solutions using web services now will lead to future web service workflows that will combine several algorithmic solutions together, and eventually build complex composite workflow solutions.

The focus of this paper is on the application of web services to image registration, and specifically to 3D medical volume reconstruction. The technology of web services opens new opportunities for applications that have high demands on computational resources (storage and computation), require setting up sophisticated computer algorithms, and involve geographically distributed expertise from multiple disciplines [3].

Generally, we have found web services to be an especially attractive technology for academic research projects because their open and interoperable nature facilitates sharing, collaboration, and discovery. This, in combination with the aforementioned advantages outweighed the risk of adopting current web service technologies [12].

Our work aims at (1) providing computational resources to end applications using web services, (2) building tools for user interaction with images (e.g., image visualization, registration feature selection), and (3) using web services for accessing sophisticated algorithms and for executing computationally intensive and memory demanding image processing queries. Our objective is to provide either a set of developed software tools or the hardware resources at NCSA or both to scientific communities with the use of web services.

In this paper, we formulate the problem of 3D volume reconstruction using I2K algorithms [8], explain the system design using web services [11], and provide a brief description of the D2K Web Service architecture [7], [10].

2. Motivation for 3D Volume Reconstruction Using Web Services

We address the problem of 3D medical volume reconstruction using web services. 3D volume reconstruction is understood as the problem of (a) mosaicking microscopy image tiles of one cross section, and (b) aligning images of multiple cross sections to form a 3D volume of large data size. The use of proposed web services is motivated by the fact that the problem of 3D medical volume reconstruction requires significant computer resources and human expertise in medical and computer science areas. We view web services as the mechanism for establishing a collaborative environment between medical and computer science collaborators and combining their geographically distributed expertise.

3. Problem Description

In a collaborative environment with medical and computer science collaborators, the goal is to reconstruct 3D medical volume from high resolution microscopy images of several cross sections. High resolution mosaic images of cross sections are formed from a large set of tiles, and then the mosaic images are aligned to construct a 3D volume. From a medical collaborator viewpoint, 3D volume reconstruction requires (a) setting up sophisticated 3D volume reconstruction algorithms and (b) computation and storage beyond the capability of a desktop computer, and therefore computer science expertise and resources. From a computer science collaborator viewpoint, 3D volume reconstruction requires selecting pairs of matching features for cross section alignment and hence medical expertise. Thus, there is need to develop a cyber-infrastructure environment where the computational resources and the expertise of remotely located medical and computer science collaborators can be integrated.

3.1. Application Scenario

We address the problem of 3D volume reconstruction in a collaborative environment by using web services. Our prototype system, shown in Figure 1, enables medical and computer science researchers to solve 3D volume reconstruction problems using web services.

The developed solution consists of the following workflow. First, a medical collaborator, e.g., from UIC, acquires images and sends data to his or her computer science collaborator. It is also possible that the medical collaborator uploads the data assuming a high bandwidth connection. Second, a computer program automatically mosaics image tiles, selects the most salient frame from each sub-volume, segments the selected frames and pre-computes centroids of all segments. The pre-processed images and centroid information are packaged for web access. Third, the medical collaborator will be notified about the URL designed for accessing and navigating the image data, as well as for selecting registration points and visualizing registration results.

The medical collaborator selects matching features for cross section alignment by using standard human computer interfaces (HCI), and our developed image navigation tools [15]. The points are saved at NCSA UIUC for additional processing after pressing the button "Compute". Pressing this button sends a query from UIC to NCSA to request registration computation. After the computation is completed, the results can be displayed by pressing the button "Result". Pressing the button "Result" executes (a) a query transmission from UIC to NCSA, (b) image transformation according to the computed parameters, and (c) a transmission of the resulting image back to UIC. In the aforementioned workflow, all operations that require intensive computation are performed at NCSA (computer resource location), while all operations that require medical domain knowledge (image acquisition, registration point selection, and volume inspection) are performed at UIC (domain expertise location).

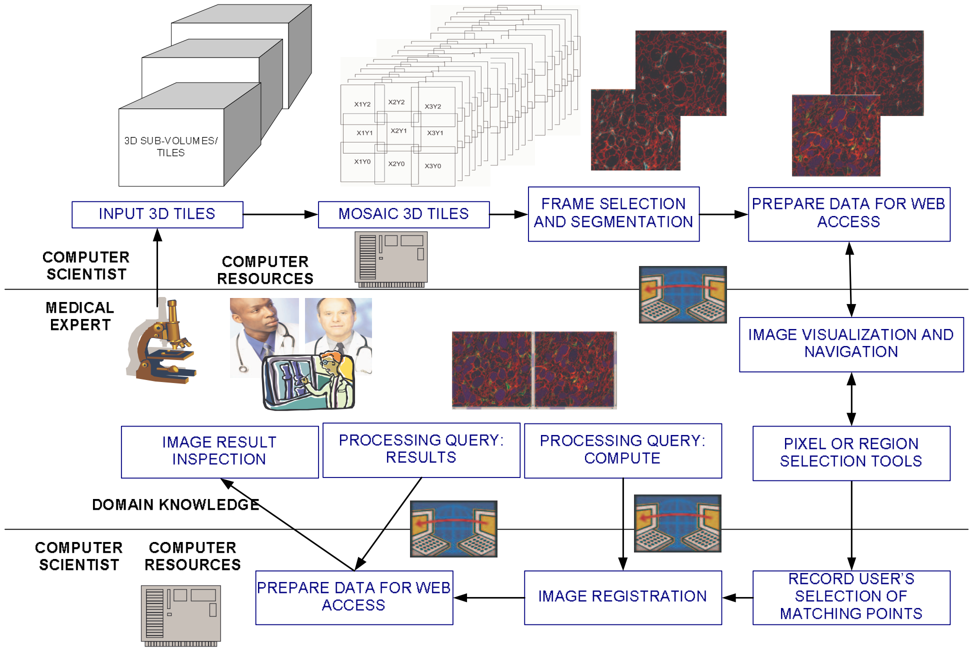

Figure 1: An overview of the current application scenario.

3.2. Input data

The input data set consists of two 3D medical volumes from the medical school at the University of Illinois at Chicago. The data are stored in TIF format. The first volume consists of (a) 21 color tiles (each tile is about 800Kbytes, 512x512 pixel size) that have to be stitched together into a frame, (b) 13 frames in each sub-volume that represents a cross section and (c) 16 sub-volumes that have to be aligned to form a 3D volume (about 5 gigabytes and about 2,750 x 1,200 x 208 pixel size). The second volume after reconstruction leads to 3.5 gigabytes and about 1,590 x 1,530 x 117 pixels.

3.3. Previous Work

There exist commercial software packages that address the problems of image access and navigation with other than web service approaches. In the medical domain, these solutions are known as “Virtual Microscopes” and primarily owned by companies (like Bacus Laboratories, Inc. [13] and Aperio Technologies [14]). In the GIS domain, the solution for accessing all IKONOS aerial photos before 9/11 was developed by Microsoft (navigation capability without annotation or computation capability).

Among the most recent solutions using web services, we should mention a new suite of web service tools to facilitate multi-sensor investigations in Earth System Science that is sponsored by NASA [4], and web services implemented for GIS data operations (ArcWeb tools [9]). The tools for NASA are developed based on a framework using grid workflows (known as SciFlo) [5]. Other workflow frameworks, like Kepler [6], have not been used for applications using web services. The ArcWeb services [9] are proprietary, and focus primarily on (a) accessing terabytes, and (b) reducing data storage and maintenance costs.

In contrast to previous work, the presented work is based on the data flow framework called D2K [7]. It is a visual programming environment and data flow execution engine developed at NCSA for data mining applications (prediction, discovery, and anomaly detection with data management and information visualization). The underlying 3D volume reconstruction algorithms came from a library of image analysis tools called I2K [8], also developed at NCSA. Our prototype system has been used in practice for 3D reconstruction of uveal melanoma tissues that is a part of the NIH-funded collaboration between UIC and NCSA.

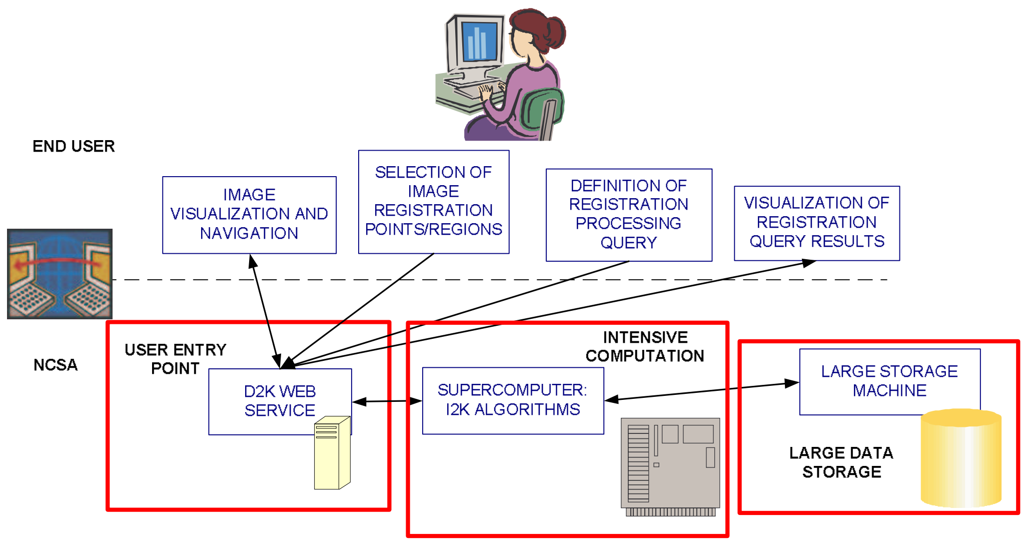

Figure 2: The overall architecture of the proposed system.

4. System Design Using Web Services

4.1. System Architecture

The overall architecture of the proposed system is illustrated in Figure 2. It is assumed that (a) all connections inside of NCSA are high speed, and (b) there might be additional high speed connections between the storage machines and the NCSA supercomputer. A user (a medical expert) will be able to interact with the system if he/she has a low bandwidth connection to the Internet. If the image size is larger than user's RAM or screen size then the image access would be enabled by (a) creating a multi-resolution image pyramid, (b) tiling images at multiple resolutions, and (c) storing pyramid and tile information in a database. The current implementation assumes that the two image frames to be registered do not exceed available memory of the medical expert's desktop.

The problem of image navigation is solved by (a) selecting and displaying image sub-areas at a chosen resolution, (b) panning through spatially large images using vertical and horizontal slider bars, and (c) selecting sub-set of bands to display to original data, pre-processed data (segmentation) and registered data. We should note that in our architecture, the user interface was implemented using Java applets. The reason for choosing Java applets comes from the fact that web services have been designed for web-based software interoperability but not for image visualization and data interaction purposes. The additional problem of registration feature selection is approached by providing tools for either pixel selection or region selection that is converted to a region centroid. The problem of intensive computation and extensive storage is tackled by (a) preparing a set of processing algorithms accessible by web services, and (b) using storage and computational resources at NCSA.

The proposed approach is based on the storage-computation paradigm, where a user is running only a "thin" client applet with small storage-computation resources and all storage demanding and computationally intensive operations are performed at NCSA. If a user would like to perform all operations at his desktop then he could install the entire system locally on his machine. We have not pursued the paradigm where data sets would be transferred to a local machine and computationally intensive processing of large size image data would be performed on a local machine.

4.2. D2K Web Service Architecture

The D2K Web Service (WS) provides a WS-I Basic Profile 1.1 compliant programming interface for executing D2K Itineraries on remote D2K Servers [7]. D2K itineraries are XML files that define data mining applications composed of D2K modules (Java classes) that have been connected together to form a directed graph. The D2K Server, like other D2K-driven applications such as the D2K Toolkit (a GUI for building itineraries), uses the D2K Infrastructure as the itinerary execution engine (see Figure 3).

Each D2K Web Service endpoint contains a library of registered itineraries available for execution by clients. For each itinerary, a pool of resources required for execution (Java classes, property files, etc.) is also stored. Associated with each itinerary definition is a list of D2K

Figure 3: D2K component architecture

Servers that are eligible to process it. When service clients submit job requests, the D2K Web Service automatically handles the brokering of execution to an appropriate, and available, D2K Server. In return, the D2K Server requests from the D2K Web Service the resources it will need to process the job. While a job is processing, the D2K Web Service monitors its progress and persists any results that are produced. All communications between the D2K Web Service and D2K Servers occur over transmission control protocol (TCP) socket connections using D2K specific protocols.

4.3 Tradeoffs of System Design

When designing the prototype we tried to optimize the system so that an end user (the medical expert) would have a responsive system that would show the most up-to-date information with a limited amount of resources used. We considered the following tradeoffs:

(1) Image data transfer: Image transfer can be executed as a transmission of the full size image vs. the use of image pyramids [1], [2], in which the system would load the image as needed, for example, when a user zoomed in/out of the image and panned around the image. In general, medical images can be large in size. In our test case, we worked with images that are approximately 1300 x 2700 pixels, but could potentially be larger. Using image pyramids will reduce the initial transfer of data but will continuously download data of the image tiles.

(2) Image segmentation computation: Segmentation of the input images can be performed on a server side or on a client side. We segment the input images in order to improve registration accuracy by replacing a pixel location with a more reliable region centroid location. The segmentation could be performed before a user selects a region (segmentation results have to be transferred) or after a user chooses a location to segment locally (computation has to occur on a client side). Segmentation on the client side is a CPU intensive task. Segmentation on the server will reduce the client CPU time but will increase the image size that needs to be transferred.

(3) Image compression: Medical image data can be transferred compressed or uncompressed. Transferring compressed input images will result in smaller data transfer, but will require some client CPU time to decompress the images. Transferring uncompressed images will increase the data transfer requirements. In our prototype, compressed images led to a size reduction from 15.3Mb to 2.27Mb.

(4) Image transformation: Computation of image transformation parameters and the transformed images can occur on a client side or on a server side. Once the expert has selected at least three points or regions in each image, we can calculate the image transformation parameters for an affine transformation and transform one image into the coordinate system of the other image. If the computation is performed on a client side, then there would be less network traffic but higher demands on the CPU usage on the client side. If it is performed on a server side, then there would be less CPU usage on a client side but more network traffic to transfer transformed image data.

In our application scenario, the objective was to create a prototype that would be very responsive to a medical expert, and it would not require significant computational resources on a client side. Thus, we tried to minimize the computer requirements for a medical expert and leverage the virtually unlimited resources of NCSA. Table 1 lists the tradeoffs we have considered during the system design. Our tradeoff considerations revolved around the following metrics: (1) CPU time on a client side, (2) RAM on a client side, (3) bandwidth for image transmission, and (4) an overhead associated with number of queries. According to Table 1, we chose the options that would limit the resources at the client (UIC) side. Our final choices were (1) to transfer full images, (2) to perform segmentation on a server side, (3) to use image compression before sending data, and (4) to compute image transformation parameters and transformed images on a server side. The images used in our prototype were small enough in size to fit into RAM memory of a standard desktop computer, and hence we decided not to use the image pyramids.

Table 1: Tradeoffs considered during a prototype system design using web services for 3D volume reconstruction. C refers to a constant, + indicates increase and – is decrease. Multiple + or – symbols indicate the magnitude of the value specified in each column.

	Client CPU	Client RAM	Bandwidth	# of Queries
Image Pyramid	C	--	++	+
Client does Segmentation	+++	-	-	C
Compression	+	+	---	C
Transformation on client side.	+++	+	-	-

4.4 Prototype Solution

The prototype solution is available at http://i2k.ncsa.uiuc.edu/MedVolume/. We have tested the prototype with the Microsoft Internet Explorer 6.0 using the Sun Java Runtime Environment (JRE) 1.4.2 browser plugin. The left lower corner of the browser conveys messages about the execution steps, e.g., Applet ncsa.uic.RegistrationApplet started. A user can select image frames to register using the drop down menu with image names. Selected images are compressed at NCSA site, transferred to a client site, decompressed and displayed in the image panels (see Figure 4).

After selecting at least three pairs of matching points/segments, the compute button is enabled and the D2K Web Service machine can be contacted. If an image panel is in the point selection mode then the left mouse click will define the pixel location to be used for registration. If an image panel is in the segment selection mode then the left mouse click will be replaced with the centroid location of the segment that contains the mouse click location. Selection of registration points using the centroid feature approach is illustrated in Figure 5.

After launching web services by pressing the button “Compute”, a new Java window labeled “Executing Job” will appear containing information about the compute job while the affine transformation parameters are computed. When the job is completed, the button “View Results” becomes active and the results can be visualized on the client side as it shown in Figure 6. The visualization will contain a seven band image that was sub-sampled in order to decrease the file size. The seven band image contains the original left image bands, the transformed right image bands, and one black band for visualization purposes.

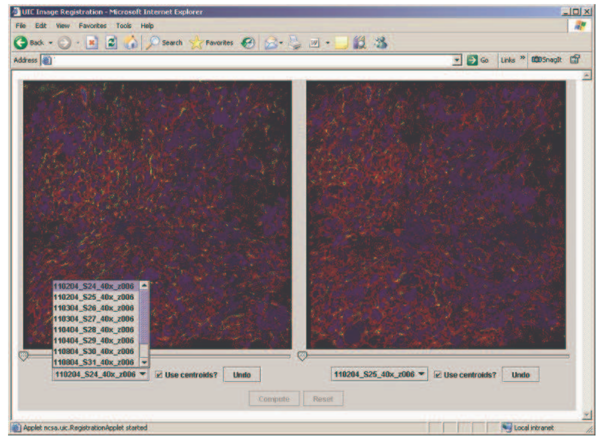

Figure 4: Image selection menu.

If the registration accuracy is satisfactory to the medical collaborators then they record the job ID number displayed in the top section of the screen. This ID number allows computer scientists at NCSA to retrieve the points associated with the correct registration session and complete the 3D volume reconstruction. Based on the set of points selected by medical experts and saved at NCSA, multiple confocal laser scanning microscope sub-volume are transformed into a reference coordinate system at NCSA and the 3D volume reconstruction results are posted for downloading at an ftp site or a web site.

5. Summary

In this paper, we presented a prototype solution for 3D medical volume reconstruction that was used in practice by UIC and NCSA collaborators. We overviewed the 3D volume reconstruction problem requirements, the architecture of the developed prototype system using web services and the tradeoffs of our system design.

In a summary, the web services based approach provides two major advantages. First, a user will be able to perform computationally intensive image operations (a) with large size image data and (b) with sophisticated 3D volume reconstruction analysis methods. Furthermore, a user will not have to invest into (a) computational and storage resources and (b) development of complicated analysis software. Second, the currently advertised interoperability feature of web services will enable us in the future (a) to customize system front end (graphical user interfaces (GUI) for the user entry point) without changing system back end (complex algorithms that perform desired computations), (b) to upgrade algorithms and fix software bugs without any involvement of a user, and (c) to integrate distributed web services that will be available on the Internet.

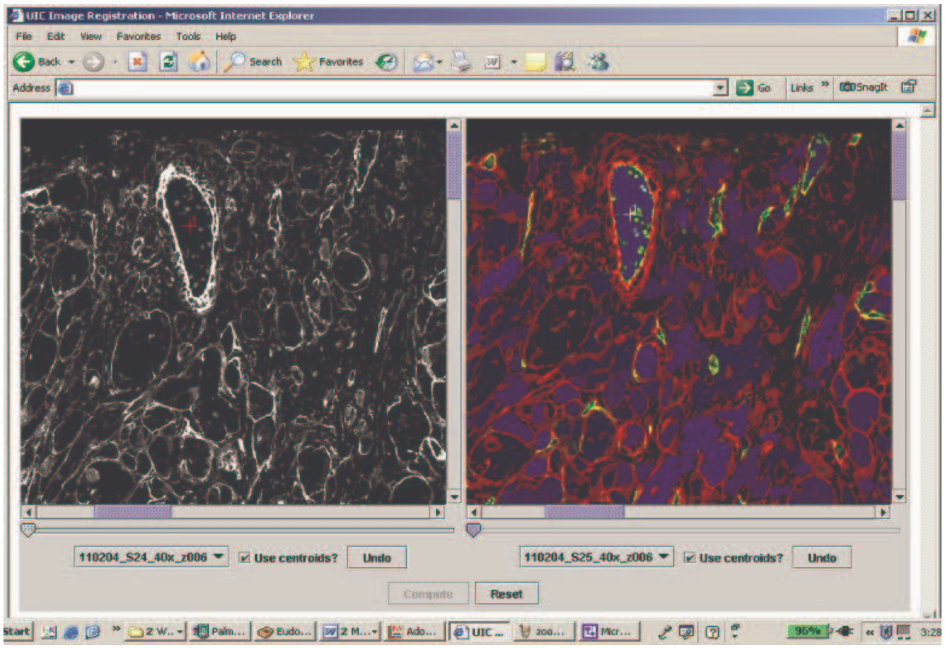

Figure 5: Selection of registration points using the centroid feature approach. Note that sub-areas the large images in Figure 4 are shown. ("+" marks represents the region centroids)

6. Acknowledgement

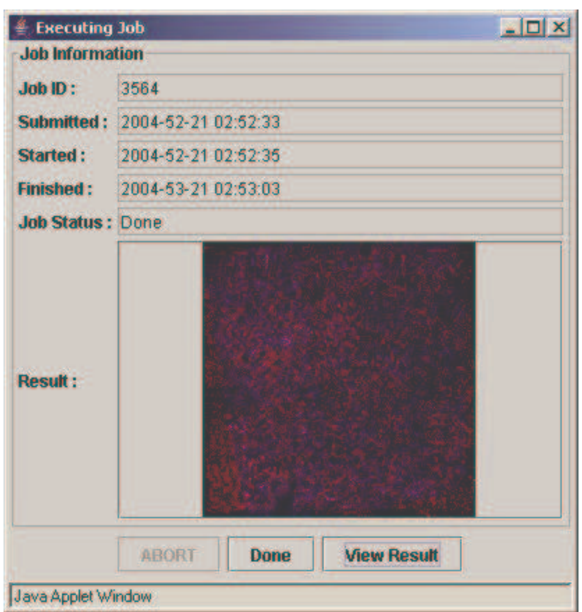

Figure 6: Job execution window after pressing the button "View Result".

This material is based upon work partially supported by the National Laboratory for Advanced Data Research (NLADR) and the National Institute of Health under Grant No. R01 EY10457. The on-going research is collaboration between the Department of Pathology, College of Medicine, University of Illinois at Chicago (UIC) and the Automated Learning Group, National Center for Supercomputing Applications (NCSA), University of Illinois at Urbana-Champaign (UIUC). We acknowledge NCSA/UIUC support of this work.

7. References:

[1] A. Rosenfeld, "Multiresolution Image Processing and Analysis." Springer-Verlag, New York 1984.
[2] V. Cantoni and S. Levialdi, "Pyramidal Systems for Computer Vision." *NATO ASI Series, Series F: Computer and Systems Sciences*, Vol. 25., Springer-Verlag, New York 1986.
[3] N. Leavitt, "Are Web Services Finally ready to Deliver?" *IEEE Computer*, Computer Society, pp. 14-18, 2004.
[4] T. Yunck, B. Wilson, A. Braverman, E. Dobinson and E. Fetzer, "GENESIS: The General Earth Science Investigation Suite" *The fourth annual NASA's Earth Technology Conference*,2004. URL:http://www.esto.nasa.gov/conferences/estc2004/papers/a1p1.pdf and http://genesis.jpl.nasa.gov/zope/GENESIS
[5] B.D. Wilson, "GENESIS SciFlo: Enabling Multi-Instrument Atmospheric Science Using Grid Workflows," *poster SF31A-0716 0800h at American Geophysical Union (AGU) Fall*

Meeting Special Focus: Advances in Data Acquisition, Management, Analysis and Display, 13-17, 2003.
[6] Kepler "An Extensible System for Scientific Workflows". (URL: http://kepler.ecoinformatics.org)
[7] "Data To Knowledge (D2K), visual programming environment description", URL:http://alg.ncsa.uiuc.edu/do/tools/d2k
[8] "Image To Knowledge (I2K), library of image analysis tools". URL:http://alg.ncsa.uiuc.edu/tools/docs/i2k/manual/index.html
[9] "ESRI ArcWeb Services" URL:http://www.esri.com/software/arcwebservices/index.html
[10] A. Shirk, "D2K Web Service Design & Implementation," *Presentation given to NCSA CyberArchitecture Working Group*, URL:http://alg.ncsa.uiuc.edu/do/documents/presentations
[11] "Web Services Interoperability Organization Basic Profile Version 1.1" URL:http://www.ws-i.org/Profiles/BasicProfile-1.1-2004-08-24.html
[12] Savas Parastatidis and Jim Webber, "Assessing the Risk and Value of Adopting Emerging and Unstable Web Services Specifications" Services Computing, 2004 IEEE International Conference on (SCC'04), 2004.
[13] WebSlide by Bacus Laboratories Inc. (URL: http://www.bacuslabs.com/indexvirtmic.html)
[14] ScanScope scanners and Digital Slide Information Management Software by Aperio Technologies. URL: http://www.aperio.com/products-software.asp
[15] S-C Lee and P. Bajcsy, "Feature based Registration of Flourescent LSCM Imagery", *in SPIE International Symposium in Medical Imaging*, 2005.