Automatic Categorization of Figures in Scientific Documents

Xiaonan Lu[1], Prasenjit Mitra[2,1], James Z. Wang[2,1], and C. Lee Giles[2,1]
[1]Department of Computer Science and Engineering and
[2]College of Information Sciences and Technology
The Pennsylvania State University
University Park, Pennsylvania, USA
xlu@cse.psu.edu, pmitra@ist.psu.edu, jwang@ist.psu.edu, giles@ist.psu.edu

ABSTRACT

Figures are very important non-textual information contained in scientific documents. Current digital libraries do not provide users tools to retrieve documents based on the information available within the figures. We propose an architecture for retrieving documents by integrating figures and other information. The initial step in enabling integrated document search is to categorize figures into a set of pre-defined types. We propose several categories of figures based on their functionalities in scholarly articles. We have developed a machine-learning-based approach for automatic categorization of figures. Both global features, such as texture, and part features, such as lines, are utilized in the architecture for discriminating among figure categories. The proposed approach has been evaluated on a testbed document set collected from the CiteSeer scientific literature digital library. Experimental evaluation has demonstrated that our algorithms can produce acceptable results for real-world use. Our tools will be integrated into a scientific-document digital library.

Categories and Subject Descriptors

H.3.3 [**Information Systems**]: Information Search and Retrieval

General Terms

Algorithms, Design

Keywords

Scientific Literature, Documents, Figures, Feature Extraction, Machine Learning

JCDL '06 June 11–15, 2006, Chapel Hill, North Carolina, USA.

1. INTRODUCTION

Figures are an integral part of documents. Especially, in scientific documents, figures are often used to illustrate the key ideas and findings, and to help readers understand the technical details of the work. For instance, statistical graphs are used to show the behaviors in response to variations of certain parameters or to compare the performance among approaches. Flow charts are used to illustrate the connections among tasks. Engineering designs are presented as diagrams or drawings. Photographs from conventional cameras or microscopes are often included in scientific documents to show readers what cannot be easily described. Human beings can interpret figures quickly and can often perceive the ideas hidden within the figures without reading the details about the figures. *"A picture is worth a thousand words."* The critical role of figures in understanding the contents of scientific documents warrants more effective use of them in scientific digital libraries.

Current digital library end users are not equipped with search engines or tools to look for information within figures. Text within the documents, including the captions of figures, are typically indexed for retrieval purposes. Ideally, search engines should use both textual and figure information to assist the users to find relevant documents.

We performed a literature survey, and found that current digital library metadata standards that define metadata elements for resource description do not include detailed information about figures. For instance, the well-known Dublin Core Metadata Initiative[1] defines fifteen metadata elements for resource description and does not include any elements describing figures contained in documents.

Additionally, tools should be developed to assist the users to make better use of the information stored in the figures for their own research. For example, if the reader is interested in comparing the performance results of their own work with that of the work presented in a document, the user should be able to quickly and automatically convert the performance plot published in the document into a numerical table of performance numbers. Scientists in certain fields often search for documents with figures containing specific experimental results. Without automated tools, post-doctoral or graduate students have to spend a lot of time to extract data from figures (i.e., the reverse process of drawing a figure from a data set)

[1]`http://dublincore.org`

in scholarly articles and insert the data into a database or spread sheets. Though biologists have been depositing DNA or protein sequence data in publicly-available databases before publishing papers, performance data shown in plots usually is not shared for later retrieval and use.

1.1 Overview of Our Work

Our goal is to automate the efforts of extracting figures from scientific documents, categorizing figures, indexing figures, and to make use of information stored in figures. For different types of figures, we plan to use different indexing and retrieval techniques. This is consistent with the semantics-sensitive approach for image retrieval [29]. For photographic figures, we can index the figures based on their color, texture, and shape. For 2-D plots, we seek to develop automated reverse-engineering tools for converting the figures into data collections that can be stored in a database.

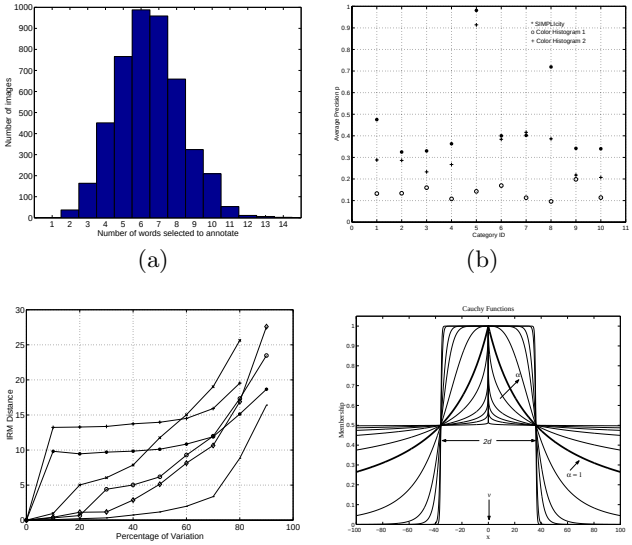


Figure 1: 2-D plots in scientific documents can appear very different. These figures were extracted from prior publications.

There are several important steps we have to take before this vision can be realized and these steps are not easy to automate. First, the figures have to be extracted from the documents. This process alone is difficult because the documents can be generated from a wide variety of hardware and software platforms, and they can be of very different qualities. Second, the figures have to be categorized automatically according to the types of the figures. There are no fixed rules for designing figures. As shown in Figure 1, even simple 2-D plots can appear very different. This diversity makes it very challenging to develop categorization algorithms. Multi-part figures, as shown in Figure 2, can be more difficult to categorize automatically because the algorithm needs to find out the number of sub-figures in a multi-part figure. Finally, we need to index the figures for effective retrieval. That means, if the figure is categorized as a 2-D plot, we need to extract data points from the plot and store them in a database system. For figures in other categories, we also will need to develop ways to index them.

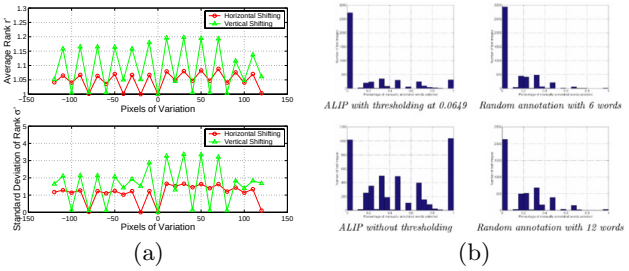


Figure 2: Multi-part figures are often more difficult to categorize than single plots. These figures were extracted from prior publications.

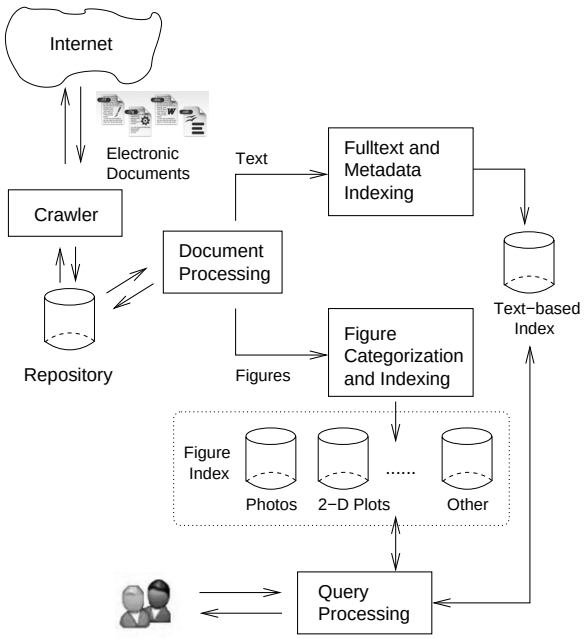


Figure 3: The architecture of our proposed document retrieval system.

We propose a system architecture, as shown in Figure 3, which combines a figure-based indexing module with full-text and metadata-based indexing modules. Besides extracting textual information, the proposed system extracts and indexes information from figures, tables, (potentially) equations, etc. We aim at designing techniques for extracting information from figures. Specifically, we propose an algorithm for automatically categorizing figures into semantic types including 'photograph', '2-D plot', '3-D plot', and 'diagram'. Different types of figures are assumed to be substantially different. Consequently, they require different techniques to further analyze and extract information for indexing.

We use a content-based approach to extract information from figures. Here, the term 'content' refers to the actual pixels in the figures rather than the semantic content. This use of the term is consistent with the field of content-based image retrieval. There are several approaches to extract information from figures within digital documents. One typical approach infers the semantics of figures from the textual information contained in the document. There are

available techniques for matching and processing textual information. For figures within a document, there are sources of textual information that may provide hints on the semantics of the figures. For example, captions of the figures, the text surrounding the figures in the document, and the text references to the figures can be used to index the figures. Text-based approaches are computationally less challenging than content-based approaches, however, they are limited in capability when handling figures. The captions of the figures may not convey the exact semantics of the figures fully. The descriptions of the figures can be difficult to locate in the document. It is impossible to find out the type of a figure or to convert it into numerical values without analyzing the content of the figure.

We propose a machine-learning-based approach to categorize figures into pre-defined semantic types using features extracted automatically from the content of figures. We designed detectable visual features that can effectively discriminate different types of figures. The features are robust enough to handle levels of noise in figures due to various imaging conditions. Automatic categorization of figures in scientific documents can be a very hard problem due to the great creativity and hence diversity shown in these figures. It is often impossible to select one set of detectable features that can be used to discriminate between all possible figures of different types. To the best of our knowledge, there is no existing work that utilizes information about figures for searching scientific literature. There exists no integrated algorithms or toolkits that can extract, categorize, and index figures in scientific documents.

Our main contributions include:

- The characterization of an important research problem, i.e., to utilize the content of figures in searching scientific literature digital libraries;
- The development of an architecture and the identification of several categories of figures in scientific documents; and
- The design and implementation of a machine-learning based algorithm for automatic categorization of figures using features extracted from figures.

The remainder of this paper is organized as follows. In Section 2, prior work in closely related areas is reviewed. In Section 3, we explain the types of figures we are to handle. We present our method for automatic figure categorization in Section 4. Specifically, techniques for extracting global and part features of figures, and the related categorization techniques, are presented. We describe the experimental setup and the results in Section 5. Finally, we conclude our work and suggest future research directions in Section 6.

2. RELATED PRIOR WORK

Because the main purpose of our work is to enable the searching of scientific literature utilizing information contained within figures, we review related work in document retrieval, document image understanding, and multimedia retrieval in digital libraries. Due to the limitation of space, we emphasize work that is most related to ours.

2.1 Document Retrieval

In recent years, machine-learning algorithms have started to be used in the development of digital library systems. There has been prior work on automatic metadata extraction from documents, as motivated by the need for interoperability. Han et al. [12] modeled the problem of metadata extraction from research articles as a classification problem. Each line in the header of a research article is classified to one class or multiple classes of the Dublin Core metadata elements. A Support Vector Machine (SVM) based method is used to classify each line of the header using linguistic features. Based on the classification results, the algorithm extracts metadata from the header lines. Hu et al. [16] proposed an algorithm to extract titles of documents using format features. Because 'author name' is a critical metadata element for document retrieval and citations, name ambiguity affects the performance of searching documents and accumulating citations. Several published articles addressed the problem of name disambiguation in digital libraries. Han et al. [13] proposed an unsupervised k-way spectral clustering method using citation attributes to disambiguate author citations. Another two-step framework was proposed [24] which combines different blocking methods and distance metrics for name disambiguation.

2.2 Document Image Understanding

According to Niyogi and Srihari [23], document image understanding aims at enabling computers to process documents, with the goal of converting an image representation of a document, for instance a paper document scanned by a flatbed document scanner, into high-level semantic descriptions of the document. There has been extensive work on many aspects of document image understanding, e.g., physical and logical structure analysis [21, 23], and indexing and retrieval of document images [8]. In tackling document image understanding problems, domain-specific knowledge is often necessary. For example, Zheng et al. [30] automated form processing by using Hidden Markov Models to detect parallel lines in a form. The parallel lines can be used to locate different parts of the form. Blostein has discussed different approaches to recognize diagrams in documents [3].

2.3 Multimedia Retrieval in Digital Libraries

Online multimedia resources, including image, video and audio libraries, have proliferated. Many different approaches have been proposed to address the problem of multimedia retrieval. There are text-based and content-based mechanisms that index multimedia, using associated text information and content (e.g., pixels in images), respectively. Christel and Conescu have compared the effectiveness of text-based search, based on transcripts and narrative text, with that of visual features for TREC video retrieval [6]. A content-based approach [27] was proposed to extract vocal signals for music data organization, retrieval and copyright protection in a digital library. Naaman et al. [22] proposed a system that can identify people in personal photo albums. Their system identifies similarities in time and location to determine events and photograph groups. Using contextual information and previous annotations, their system identifies people in photographs.

Since the 1990s, content-based image retrieval has attracted a lot of attention. A review article by Smeulders et al. [25] provided details related to the history, scope, content description, and semantic interpretation problems. A follow-up survey by Datta et al. analyzed the post-2000 progress in the field [7].

Our work is a type of specialized content-based image retrieval focusing on figures contained in scientific documents. Using domain-specific knowledge about scientific literature, algorithms can be developed to extract useful information from figures. Our work is closely related to document image understanding. However, in our work we deal with individual figures rather than scanned pages of the documents. Every figure, once extracted from the document, contains an independent object. In document image understanding, an image is typically a page that has a logical structure and can contain different types of objects including paragraphs, figures, and tables.

3. CATEGORIES OF FIGURES

The goal of our work is to obtain useful information from figures and to integrate figures into scientific literature search. We first have to define semantic types of figures in a way that facilitates further analysis of figures. We take the functionalities of the figures into consideration in defining semantic types for figures. The techniques for extracting information from figures depend on the functionalities. As an example, the number of curves and the data contained in each curve can be of interest in figures containing 2-D data plots. In figures that contain diagrams, the organization of the diagrams and the number and the flow of the modules (i.e., those represented by rectangles and circles), are what we are interested in when we attempt to decipher the figures.

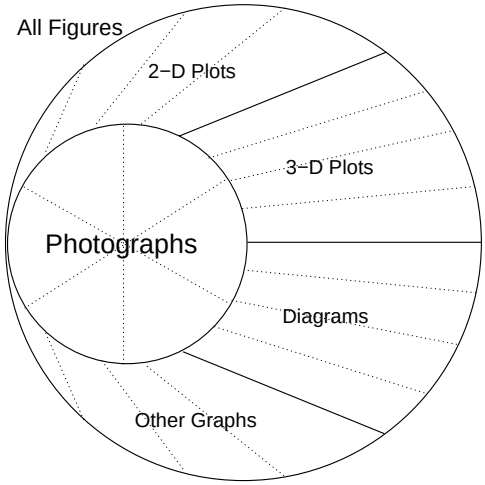


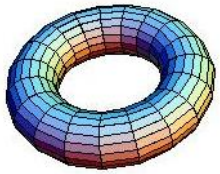
Figure 4: The initial hierarchy of figure categories. A figure is categorized as a photograph or a non-photograph. A non-photograph is then further categorized. Dotted lines indicate possible sub-categorization in the future.

We use two rules to guide our process of defining the semantic types of figures: (1) A semantic type of figures serves a specific purpose; and (2) Figures of a semantic type share some degree of visual similarity that can potentially be captured computationally. Both these rules are important. We cannot develop algorithms to categorize figures unless there are some visual features we can use to distinguish among different types. On the other hand, the categorization is going to be useful only if the functionalities of figures in the same semantic type are similar.

In our study, we used documents crawled from the Web by the CiteSeer [11] scientific literature digital library. CiteSeer focuses on literature in computer and information sciences and engineering. Based on a study of more than five thousand figures contained in research papers on CiteSeer, we define an initial hierarchy of semantic types for figures, based on functionalities in the documents and the visual appearances. We observed that some types of figures appear in papers more frequently than others. For example, in our dataset, photographic figures and 2-D plot figures are used more than 3-D plots and diagrams.

An initial hierarchical structure, as shown in Figure 4, is defined to represent relationships among different categories of figures. In the rest of this section, we will provide a definition for each semantic type of figures based on their functionalities in the documents. We would like to emphasize that this hierarchy is preliminary. Ideally, the categorization can be much more complete by studying figures from various fields, including biology, medicine, physics, chemistry, other engineering sciences, and humanities.

(a) Photo



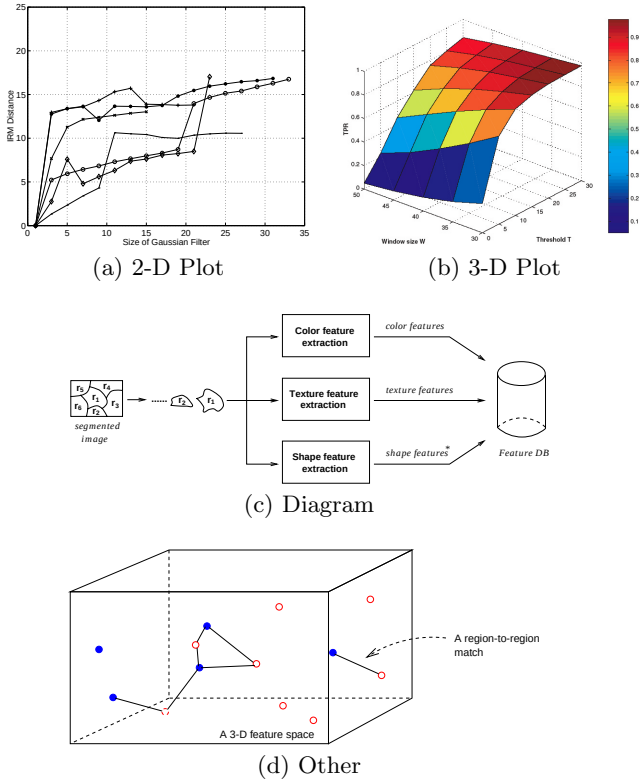
(b) Computer Graphics

Figure 5: Some figures categorized as "photograph" figures because they include continuous tones. Other examples include microscopic slides and computed tomography images.

Photograph: A continuous-tone figure recorded by a camera or created by photo processing software. This is consistent with the definition given by Li and Gray [19]. Pictures generated by computer graphics techniques, e.g., dinosaur shots from the movies, usually fall into this category. Similarly, images of pathological tissue taken under a microscope or computed tomography images are considered photographs. Figure 5 shows some example figures categorized as photographs. The further categorization of photographs is becoming a main problem in the image retrieval field [20, 7].

Non-photograph: A non-continuous-tone image, as defined by Li and Gray [19]. That is, a figure is either categorized as a photograph or a non-photograph. Examples of this category are included in Figure 6.

2-D Plot: A non-continuous-tone image that contains a two-dimensional coordinate system (i.e., horizontal and vertical axes) and a series of points, lines, curves, or areas that represent the variation of a variable in comparison with



(a) 2-D Plot (b) 3-D Plot

(c) Diagram

(d) Other

Figure 6: Some example non-photograph figures extracted from prior publications.

that of another variable. Examples of 2-D plot are scatter plots, curves, and histogram bar charts. Figure 1 shows some example 2-D plots.

3-D Plot: A non-continuous-tone image that contains a three-dimensional coordinate system and a series of points, lines, curves, or areas that represent the variation of a variable in comparison with that of the other two variables. Note that some 3-D plots may be printed with continuous tones. In that case, according to our categorization, they can be categorized as photographs.

Diagram: A non-continuous-tone image that shows arrangements and relational dependencies among a series of components. Components in the diagram are usually represented by closed contours such as: rectangles, ovals, diamonds, etc. Relational dependencies among components are represented by lines, arrows, etc. An Entity-Relationship diagram for modeling a database application is a typical example.

Others: A non-continuous-tone image that does not belong to any of the above classes. New categories of figures will be created out of this class in the future. Some examples are pie charts and figures with subfigures (Figure 2).

Currently, there are five leaf nodes in this hierarchical structure corresponding to the figure types we attempt to categorize. In the future, we will define more categories of figures depending on user requirements and the potential for successful algorithm development.

4. THE METHOD

In this section, we first present a image preprocessing technique that extracts figures contained in documents and store them in an image format. Then, we present the image features we have used, the reason behind selecting each feature, and the techniques for extracting the features. Finally, we present the categorization techniques.

Before the figures can be analyzed, categorized, and indexed, we must first extract figures from the documents and store them in a common image format. Next, we propose an algorithm to categorize the figures based on visual features. The algorithm utilizes both global features and part features [26] in order to capture the global patterns of figures as well as properties of specific objects in figures. A *global feature* refers to a global property of an image, e.g., the average gray level or the texture. A *part feature* refers to a part of an image with some special properties, e.g., a circle or a line.

4.1 Image Preprocessing

There exist different approaches for extracting figure images from digital documents. One way is to process the original document file directly. Another way is to convert every page of a document into an image file and extract figures using document image understanding techniques [1]. The first approach is suitable for documents generated directly by a typesetting system, while the second is necessary for scanned documents. In our case, we assume the documents are converted directly into the PDF format using a typesetting software such as Microsoft Office or LaTeX. We use some off-the-shelf tools, e.g., Adobe Acrobat, to extract the figures from the PDF files.

After figures have been extracted from documents, they are converted to the Portable Gray Map (PGM) format, a grayscale image format, which is easy to manipulate. Note that original images extracted from documents can be color images. Typically we do not need to see the figures in full color in order to determine the semantic type of a figure, though full color certainly helps in understanding the meanings of the figure. Thus, we convert all images to grayscale format in order to save computation time and memory consumption during content analysis. If in some particular cases, color information is necessary for the analysis of figures, the original color images corresponding to the figures can be easily loaded from the image database.

We crop out boundaries of ten pixels on each side. This is because some figures have boundary boxes which can be a problem in the automated categorization process.

4.2 Global Image Features

We use global image features to discriminate photographs from other images. Intuitively, the overall information of an image needs to be analyzed before it can be categorized as a photograph or non-photograph. Li and Gray [19] have developed a highly robust algorithm to segment images into picture, text, and background regions. Their algorithm outputs three global image features: the percentage of the figure that are picture, text, and background areas. Clearly, the sum of the three global features equals one hundred percent because we assume that every single image region belongs to one and only one of those three classes. One of the advantages of choosing these three global image features lies in its flexibility of dealing with various kinds of figures

contained in documents. For example, a figure with majority text and background, and a small area that can be classified as a picture region, is expected to be categorized as non-photograph based on the values of the three global image features.

For the benefit of readers, we now provide a brief explanation of the algorithm developed by Li and Gray [19]. Every figure is divided into non-overlapping small blocks. The user can choose the size of the image block based on the needs of the application. The wavelet transform is performed on the figure and coefficients are localized to every block. Vetterli and Kovacevic have reported that wavelet coefficients of photographs in the high frequency bands tend to follow a Laplacian distribution [28]. The goodness of fit between the distribution of wavelet coefficients of every image block and the Laplacian distribution is calculated. Besides, the likelihood of the wavelet coefficients being composed of highly concentrated values is calculated because the histogram of wavelet coefficients in a text block tends to have several concentrated values while that of a photograph does not. Combining these two values using a weighted sum function, a final function value is calculated for every image block, and the image block is categorized into one of the three classes: picture, text, and background. Experiments reported in [19] show the effectiveness and robustness of this method in categorizing image regions.

Having the image segmentation result, global image features are calculated based on class labels of image blocks contained in the image. Specifically, the percentage of picture, text, and background areas in relation to the image area is calculated as the number of blocks belonging to each class divided by the total number of blocks in the image respectively. That is,

$$f_1 = \frac{\#\ of\ picture\ blocks}{\#\ of\ blocks},\ f_2 = \frac{\#\ of\ text\ blocks}{\#\ of\ blocks},$$

$$f_3 = \frac{\#\ of\ background\ blocks}{\#\ of\ blocks}\ .$$

By thresholding on these three features, we can determine if a figure is a photograph or non-photograph. This process has been used in the SIMPLIcity system for image retrieval [29].

4.3 Part Image Features

We now describe the features designed for discriminating different types of non-photograph figures as well as the techniques for extracting those features.

Based on the definitions of non-photograph image types and the study of non-photograph figures contained in scientific documents, we observe that non-photograph images are mostly computer-generated graphics. They can be characterized by different objects contained in the figure. For instance, lines, curves, arrows, rectangles, ovals, and diamonds are common objects in non-photograph figures. According to the recognition-by-components theory [2], a theory of recognition in human beings, the human visual system extracts geons (i.e., geometric ions) and uses them to identify objects. There are only a small number of (less than 36) geons. Any unique object is defined by specific geons and their positions.

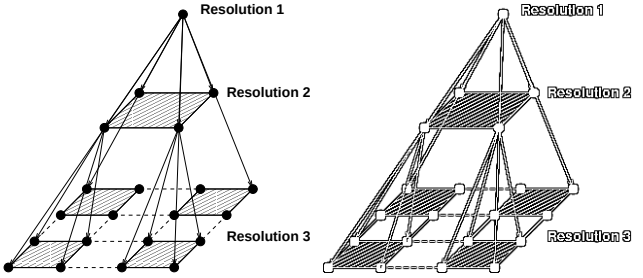
Inspired by this recognition-by-components theory, we attempt to enable computers to "recognize" different classes of non-photograph figures by automatically detecting and locating visual "geons" contained in figures. We start with a basic and common visual component in non-photograph figures, that is, straight lines. Because these basic visual components targeted are parts of the figure with certain special properties, the features associated with the visual component are part-image features based on the definition of part image features stated earlier.

4.3.1 Edge Detection

Before detecting and locating basic visual components in non-photograph figures, edge detection is applied to the PGM image to generate corresponding binary edge images. Edges refer to pixels at or around which the image values undergo a sharp variation [26]. Edges in a figure can tell us the shape, size, and position of the object if properly used.

One important reason for applying edge detection is that the commonly used line detection algorithm, the Hough transform [9], takes binary images as input.

We have used the Canny edge detector [4], likely the most well known and commonly used edge detector, to extract edges in non-photograph figures. For clarity of presentation, we briefly explain the basic techniques involved in Canny edge detection here. The process involves three steps: noise smoothing, edge enhancement, and edge localization [4]. Noise smoothing aims at suppressing image noise via filtering without destroying the true edges. After noise smoothing, edge enhancement aims to design a filter that produces a large output at edge pixels and low output elsewhere. The final step, edge localization, involves thinning and thresholding, which suppress non-essential edge points from the edge map. The resulting edge lines are better connected than some simpler edge detection algorithms.



(a) Original Figure (b) Edge Map

Figure 7: Effect of Canny edge detection. The original figure was extracted from prior publication. Some artifacts can be introduced in the edge detection process.

After edge detection, a new binary image (the edge map) is generated for every figure. The edge map has the same spatial dimensions as the original image and preserves detected edge information. In order to demonstrate the effect of Canny edge detection on the figures contained in scientific documents, the original image and the generated edge image of a random figure are shown in Figure 7. A black pixel in the edge map indicates an edge point, while a white pixel indicates a non-edge point.

4.3.2 Line Features

We detect straight lines using the Hough transform [9] and features of the detected straight lines are extracted as part image features. In every edge image, properties of the longest lines (positions of two ending points) are extracted as part features. This process aims at extracting coordinate axes commonly seen in 2-D plots.

Besides the Hough transform, template matching, a convolution-based technique, detects straight lines in the image itself. However, in order to detect lines with arbitrary orientation, a large number of masks need to be designed and used in the convolution process. Besides, image noise due to various imaging conditions makes it difficult. Based on our observation, the basic visual components including straight lines are often sparsely distributed. Many non-photograph figures are noisy. The Hough transform is selected because it is particularly useful when the patterns targeted are sparsely distributed in the image and the image is noisy.

The Hough transform has become a standard tool to locate simple patterns such as lines and ovals in an image by analyzing a relatively small area (such as a point) in the transformed parameter space. Basic visual components, including lines, curves and contours, that characterize typical non-photograph figures are almost always in forms of edges and thus preserved in the corresponding edge map obtained from the edge detection. After edge detection, the remaining information in the edge map is sufficient in general for categorizing non-photograph figures. If needed in the future when we categorize the figures into more types, we may choose to use non-edge information from the original figure.

4.4 Categorization

We use a supervised learning based approach to categorize different types of figures in scientific documents using extracted global and part image features. For our problem, the goal is to find the optimal boundaries among different categories of figures in the feature space. A machine-learning-based approach is naturally suitable and easier to implement than a rule-based method.

There are different categories of approaches for supervised learning. According to [14], one category of pattern recognition approaches, the SVM, has clear connection to the underlying statistical learning theory, which includes regression estimation, linear operator inversion, and the generalized technique. Another category of methods are characterized by learning from examples, e.g., neural networks. Neural networks are suited for applications where there is no interpretation for prediction and no interpretation of roles of individual inputs. Scalability is one of the major limitations of neural networks. Because our problem has clear connection to statistical learning theory, we chose SVM. Besides, SVM has good generalization performance and ability to handle high dimensional data [17]. These properties are important because we expect many more features, such as all lines and curves, to be included in the future to identify sub-categories of figures.

5. EXPERIMENTAL RESULTS

Our dataset is collected from the CiteSeer [11] scientific literature digital library. As mentioned before, CiteSeer focuses on literature in computer and information sciences and engineering. We collected research papers in the PDF format. About two thousand PDF files are randomly selected from the CiteSeer digital library in the experiments.

We have implemented an experimental system for automatic extraction and categorization of figures in scientific documents, as shown in Figure 8. The system has image preprocessing, global feature extraction, edge detection, part feature extraction, data preparation, and classification modules.

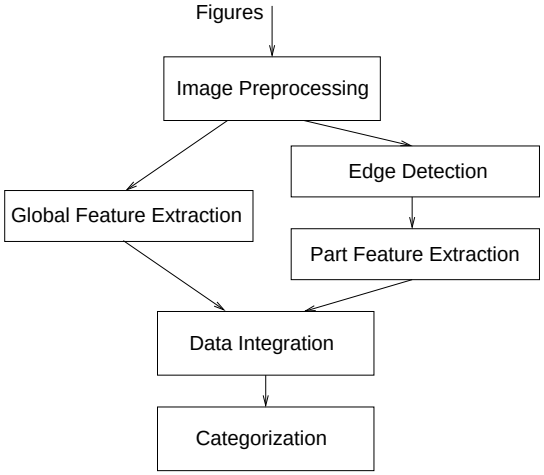


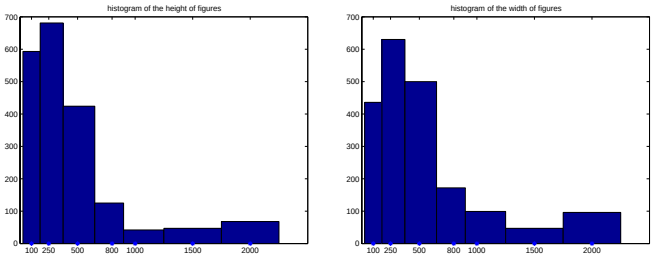
Figure 8: The flow of the figures through the components of the experimental categorization system.

We briefly introduce the functions of each module in the system in the order of data flow. As the starting point, the image preprocessing module extracts figures from digital documents, converts extracted images to grayscale format, and manages manual annotations of figures. After the preprocessing step, figures contained in our dataset are stored in the PGM format. Manual annotations and other properties of figures are stored in a database and managed using the MySQL engine. The feature extraction modules extract the global and part features, as discussed before. The data organization module manages feature data for the machine learning process. The classification module uses the SVM Light package [17] to categorize figures into the predefined semantic classes.

5.1 Experiments

The Adobe Acrobat image extraction tool is used to extract figures from documents. The spatial dimensions of extracted figures have been determined and recorded. Distributions of the heights and widths of extracted images are provided as information for the readers in Figure 9.

After extracting figures from the dataset, we manually categorize figures — required for the training and verification of our automated algorithms — to the pre-defined semantic categories: photograph, 2-D plot, 3-D plot, diagram, and others. We noticed that the Adobe Acrobat image extraction tool breaks some figures into small parts when extracting images from PDF documents. Currently, we categorize these invalid figures into the category 'others'. Besides, we realized possible ambiguity in the categorization of figures in our dataset. For instance, some 3-D plot



(a) Figure Height (b) Figure Width

Figure 9: The distributions of the figure dimensions. X: number of pixels. Y: number of images.

figures that appear to have continuous tones are categorized as 'photograph' based on the definition. The number of figures in every semantic class as well as the corresponding percentage are shown in Table 1.

Table 1: Composition of the figure dataset

Category	# of Figures	Percentage
Photograph	460	23%
2-D plot	341	17%
3-D plot	20	1%
Diagram	93	5%
Others	1066	54%
Total	1980	100%

As can be seen, photograph and 2-D plot figures appear much more frequently than diagrams and 3-D plots in our dataset. Note that since our dataset is collected from the CiteSeer scientific literature digital library, the distribution gives us information on the different categories of figures in computer and information science papers. The distribution of figures among the types may be very different if we use datasets from other fields or use a different random set. It is likely that every field has its own needs, presentation styles, and biases toward using different types of figures.

In the process of extracting global texture features, every image is divided into blocks of 8×8 pixels. As mentioned before, every block is classified as a photograph, text, or background block using the distribution of its wavelet coefficients. The three features f_1, f_2, f_3 are computed by the system developed by Li and Gray.

Before we extract line features from contents of images, Canny edge detection is applied. We have tested various parameter settings of the Canny edge detector for our figure images. The parameters we used are: (1) The standard deviation of Gaussian smoothing filter is set to 0.6; (2) The high value to be used in hysteresis thresholding is set to 0.8, which specifies the percentage point in the histogram of magnitude of gradient; and (3) The low value to be used in hysteresis thresholding is set to 0.3, which specifies the fraction of the computed high threshold edge-strength value. The performance of the detector is not very sensitive to the parameters in our application.

Three categorization problems have been tested on our dataset: (1) 'photograph' versus 'non-photograph'; (2) five semantic classes of figures: 'photograph', '2-D plot', '3-D plot', 'diagram', and 'others'; and (3) '2-D plot' versus other 'non-photograph'.

For the first test, only global features are used. In the second test, both global features and part features (line features) are used because we need to test the effectiveness of combined features on discriminating all semantic classes. In the third experiment, we used line features and heuristics.

5.2 Performance Measures

We use classification error rate (defined below) to measure the performance of categorization. Besides, precision and recall are used to measure the performance of retrieving a specific class. Classification error rate is defined as the ratio of the number of figures misclassified and the total number of figures tested. For a single figure type, we denote A as the number of figures correctly categorized by the system to the figure type, B as the number of figures categorized by the system to the type, and C as the number of figures categorized by human in the type. The precision and recall of the categorization for the type are $precision = A/B$ and $recall = A/C$, respectively. Precision measures the percentage of correctly categorized figures in relation to the number of figures categorized by a computer to the type. Recall measures the percentage of the figures correctly categorized in each type, compared with human categorization.

We determine both the training error rate and test error rate. For training error rate, the dataset serves as both the training data and the test data. In order to determine the test error rate, we use the five-fold cross-validation [14] method to estimate the generalization error. Specifically, the whole dataset was divided into five subsets of equal size. Training and classification processes are conducted five times. At each time, one subset was chosen as the test set and the remaining subsets form the corresponding training set. Finally, the overall test error rate is calculated as the average of five test error rates.

5.3 Results

5.3.1 Photograph vs. Non-photograph

We used SVM Light [17] to categorize photograph and non-photograph figures based on the global image features. Under default parameters, results are reported in Table 2.

Table 2: Precision and recall of photograph vs. non-photograph categorization based on global image features

Category	# of images	Precision	Recall
Photograph	460	68.3%	77.8%
Non-photograph	1520	93.0%	89.0%

As can be seen from the result, global image features extracted with the Li and Gray system are fairly effective in discriminating photograph vs. non-photograph figures.

5.3.2 Multi-class Problem

We used the SVM Multiclass package [17] to categorize figures into five semantic types using both global and part image features. We tested the overall multi-class categorization error rate first. We have adjusted one learning parameter, trade-off between training error and

margin, by setting it to 1.0 based on our experimental results. The training and test error rate for the multi-class problem are shown in Table 3.

Table 3: Multi-class categorization error rate

Training error rate	Test error rate
38%	40%
57%	54%

Besides the multi-class categorization error rate, we also look into the classification performance for every single class, which is useful for applications that aim to retrieve a specific class.

Precision and recall for every class are presented in Table 4. In addition, the confusion matrix that contains the full distribution of the results for this multi-class categorization problem, is shown in Table 5. Each column of the matrix represents the figures in a predicted category. Each row represents the figures in the actual category. The results show that precision and recall for photograph and 2-D plot figures are much higher than those for 3-D plot and diagram figures. The relative effectiveness of combined features on discriminating photograph and 2-D plots can be seen. Even though line features show their potential for certain types of figures, e.g., 2-D plot figures, they are not working well by themselves. It is essential to design more effective image features that have the capability of effectively categorizing the figures.

Table 4: Precision and recall of multi-class categorization

Category	# of images	Precision	Recall
Photograph	460	59%	82%
2-D plot	341	31%	55%
3-D plot	20	2%	15%
Diagram	93	8%	20%
Others	1066	79%	24%

Table 5: Confusion matrix of multi-class categorization. Column: predicted category. Row: actual category.

	Photo	2-D	3-D	Diagram	Others
Photo	375	33	6	8	38
2-D plot	40	188	27	64	22
3-D plot	0	15	3	2	0
Diagram	13	42	8	19	11
Others	212	345	98	151	260

In our dataset, the percentage of 3-D plot figures is only one percent. The percentage of diagram figures is five percent. These low percentages can possibly make the performance on these two semantic classes unreliable. Lack of enough training samples for these two classes is a problem that we plan to address in the future.

5.3.3 2-D Plot vs. Other Non-Photograph

The precision and recall of recognizing 2-D plots is shown in Table 6. The recall is high because the line properties of the 2-D coordinate system are quite useful. Adding more features and more heuristic rules to the process may improve the performance.

Table 6: Precision and recall of retrieving 2-D plot from non-photograph figures

	Precision	Recall
2-D plot	81%	82%

6. CONCLUSIONS AND FUTURE WORK

We have argued that in scientific digital libraries it is essential to extract and categorize figures, and to index the figure content so that they can be used for efficient information retrieval. We proposed types of figures that commonly occur in the scientific literature. We outlined algorithms that enable us to classify the figures into five types. Our empirical results showed that acceptable precision and recall can be achieved by using supervised learning on global and part features.

In the future, we plan to work on the following directions. For the image preprocessing module, we will look for more reliable techniques to extract figures from PDF files. In order to achieve better representation of different semantic types of figures, we plan to design and test more part image features including rectangle, curve, and ovals. Besides, we plan to combine our content-based approach with a text-based approach for analyzing the figures for retrieval purposes. We will explore the possibility of extracting numerical values from 2-D plots. In the experiments, we plan to include datasets consisting of research papers from a wide range of research areas. Finally, we plan to integrate the classification results with current full-text and metadata indexing for scientific literature search.

7. ACKNOWLEDGMENTS

This work was supported in part by the US National Science Foundation under grants 0535656, 0347148, 0454052, and 0202007, and Microsoft Research. The authors would like to thank Jia Li for providing the source code for the classification of photographs and graphics [19]. We used the Canny edge detection program [15] and the SVM Light package available in the public domain. The comments and constructive suggestions from anonymous reviewers and Edward A. Fox have been very helpful in our revision of the manuscript.

Due to copyright concerns, some figure images used in our illustrations were extracted from our own prior publications. Specifically, Figures 1(a) and 2(b) were published in [20]. Figures 1(b), (c), and 6(a), (c), (d) were published in [29]. Figures 1(d) and 2(a) were published in [5]. Figure 5(a) was created by James Z. Wang. Figure 5(b) was published in [18]. Figure 6(b) was published in [10]. Figure 7(a) was published in [20].

The techniques we have developed are intended for practical use in digital libraries. When copyright is a potential issue, readers are advised to be vigilant when deciding if it is suitable to use this technology.

8. REFERENCES

[1] H. S. Baird, D. Lopresti, B. D. Davison, and W. M. Pottenger. Robust document image understanding technologies. In *HDP '04: Proceedings of the 1st ACM Workshop on Hardcopy Document Processing*, pages 9–14, 2004.

[2] I. Biederman. Recognition-by-components: A theory of human image understanding. *Psychological Review*, 94:115–147, 1987.

[3] D. Blostein, E. Lank, and R. Zanibbi. Treatment of diagrams in document image analysis. In *Diagrams '00: Proceedings of the First International Conference on Theory and Application of Diagrams*, pages 330–344, 2000.

[4] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.

[5] Y. Chen and J. Z. Wang. A region-based fuzzy feature matching approach to content-based image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(9):1252–1267, 2002.

[6] M. G. Christel and R. M. Conescu. Addressing the challenge of visual information access from digital image and video libraries. In *JCDL '05: Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries*, pages 69–78, 2005.

[7] R. Datta, J. Li, and J. Z. Wang. Content-based image retrieval - approaches and trends of the new age. In *MIR '05: Proceedings of the 7th International Workshop on Multimedia Information Retrieval*, pages 253–262, 2005.

[8] D. Doermann. The indexing and retrieval of document images: A survey. *Computer Vision and Image Understanding*, 70(3):287–298, 1998.

[9] R. O. Duda and P. E. Hart. Use of the Hough transformation to detect lines and curves in pictures. *Communications of ACM*, 15:11–15, 1972.

[10] E. Giladi, M. G. Walker, J. Z. Wang, and W. Volkmuth. SST: An algorithm for finding near-exact sequence matches in time proportional to the logarithm of the database size. *Bioinformatics*, 18(6):873–879, 2002.

[11] C. L. Giles, K. Bollacker, and S. Lawrence. CiteSeer: An automatic citation indexing system. In *Proceedings of the Third ACM Conference on Digital Libraries*, pages 89–98, 1998.

[12] H. Han, C. L. Giles, E. Manavoglu, H. Zha, Z. Zhang, and E. A. Fox. Automatic document metadata extraction using support vector machines. In *JCDL '03: Proceedings of the 3rd ACM/IEEE-CS Joint Conference on Digital Libraries*, pages 37–48, 2003.

[13] H. Han, H. Zha, and C. L. Giles. Name disambiguation in author citations using a k-way spectral clustering method. In *JCDL '05: Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries*, pages 334–343, 2005.

[14] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer-Verlag, New York, NY, 2001.

[15] M. Heath, S. Sarkar, T. Sanocki, and K. Bowyer. Comparison of edge detectors: a methodology and initial study. *Computer Vision and Image Understanding*, 69(1):38–54, 1998.

[16] Y. Hu, H. Li, Y. Cao, D. Meyerzon, and Q. Zheng. Automatic extraction of titles from general documents using machine learning. In *JCDL '05: Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries*, pages 145–154, 2005.

[17] T. Joachims. *Making Large-Scale Support Vector Machine Learning Practical*. MIT Press, Cambridge, MA, 1998.

[18] D. Joshi, J. Li, and J. Z. Wang. A computationally efficient approach to the estimation of two- and three-dimensional hidden markov models. *IEEE Transactions on Image Processing*, 2006, to appear.

[19] J. Li and R. M. Gray. Context-based multiscale classification of document images using wavelet coefficient distributions. *IEEE Transactions on Image Processing*, 9(9):1604–1616, 2000.

[20] J. Li and J. Z. Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(9):1075–1088, 2003.

[21] S. Mao and A. Rosenfeld. Document structure analysis algorithms: a literature survey. In *Proceedings of SPIE*, pages 197–207, 2003.

[22] M. Naaman, R. B. Yeh, H. Garcia-Molina, and A. Paepcke. Leveraging context to resolve identity in photo albums. In *JCDL '05: Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries*, pages 178–187, 2005.

[23] D. Niyogi and S. N. Srihari. Knowledge-based derivation of document logical structure. In *Proceedings of the Int. Conference on Document Analysis and Recognition*, pages 472–475, 1995.

[24] B.-W. On, D. Lee, J. Kang, and P. Mitra. Comparative study of name disambiguation problem using a scalable blocking-based framework. In *JCDL '05: Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries*, pages 344–353, 2005.

[25] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(12):1349–1380, 2000.

[26] E. Trucco and A. Verri. *Introductory Techniques for 3-D Computer Vision*. Prentice-Hall, Upper Saddle River, NJ, 1998.

[27] W.-H. Tsai and H.-M. Wang. On the extraction of vocal-related information to facilitate the management of popular music collections. In *JCDL '05: Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries*, pages 197–206, 2005.

[28] M. Vetterli and J. Kovacevic. *Wavelet and Subband Coding*. Prentice Hall, Englewood Cliffs, NJ, 1995.

[29] J. Z. Wang, J. Li, and G. Wiederhold. SIMPLIcity: Semantics-sensitive integrated matching for picture libraries. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(9):947–963, 2001.

[30] Y. Zheng, H. Li, and D. Doermann. A parallel-line detection algorithm based on hmm decoding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(5):777–792, 2005.