Automatic Document Genre Identification For Faceted Document Browsing And Searching

要　旨

企業内の大規模なリポジトリ中のドキュメントを閲覧し検索する手法が大きな課題になってきている。インターネットの検索結果にユーザはほぼ満足している一方で、企業内におけるドキュメント検索にはデータタイプやユーザの検索要求の違いのためにまだ改善の余地がある。我々は、企業内のリポジトリの階層構造を保持したまま、文書ジャンルという新しい属性を含むファセットにより文書を閲覧・検索が可能なシステムを提案する。まず、電子文書・スキャン画像内の必要な情報の検索支援のため、文書の画像特徴から、技術文書、プレゼン資料、表、写真等の文書ジャンルを自動的に検出するソフトウエアを開発した。次に、選択されたファセットと検索条件にマッチした文書だけを表示するように文書とディレクトリをフィルタし、可能性の高いディレクトリを強調表示する。これらにより、ファイルの拡張子にかかわらずに、プレゼンテーションスライドだけを閲覧・検索すること等が可能となる。さらに、サムネールと自動的に抽出されたキーフレーズが必要な文書の選択を支援する。

Abstract

Browsing and searching documents in large enterprise document repositories is an increasingly common problem. While users are usually satisfied with Internet search results, enterprise search has not been as successful because of differences in data types and user requirements. To support users in finding desired information from electronic and scanned documents, we created an automatic detector for genres such as papers, slides, tables, and photos based on imaged document features. The automatically identified genres play an important role in our faceted document browsing and search system. The system presents documents in a hierarchy as typically found in enterprise document collections. Documents and directories are filtered to show only documents matching selected facets and containing optional query terms and to highlight promising directories. Thumbnail images and automatically identified keyphrases help select desired documents.

Author

Francine Chen
Andreas Girgensohn
Lynn Wilcox

FX Palo Alto Laboratory, Inc

1. Introduction

Enterprise search has been defined in different ways, including search of an organization's intranet, search of an organization's external website, and search of any text content in electronic form, such as email and databases [4]. In this paper we focus on search of unstructured information in a corporate document repository. In this type of enterprise search, an employee typically knows or remembers some attributes of the target results [10]. We describe two typical examples of this. An employee needs to find information from a presentation he saw at a project review several months ago. He does not know where the presentation material is located, who gave the presentation, or even which organization in the company created the material. He does know he saw the information in table format in a slide presentation last spring. Another employee is working on a solving a problem in product design and remembers a similar problem that was solved in another product several years ago. She would like to find information on the solution to the problem, including how it was solved and who solved it. Since the product manager has since left the company, she needs to find this information by search. She knows the product name, the rough time frame, and the nature of the problem.

Locating documents in an enterprise context most often involves users finding specific documents that either they created or they know or expect were created. In these activities, the user's knowledge of the organization, its history, and its policies and practices can be valuable in helping to locate the desired documents.

This is in contrast to Internet users who are searching among a set of documents of which they have little prior knowledge. Additionally Internet users are often searching for a fact, such as the phone number for the local restaurant, or a general discussion of a topic, such as what is happening with a particular politician. As a result of these goals, many different documents meet their needs. The restaurant phone number could be found on a restaurant review site, a Yellow Pages site, in addition to the restaurant's web site. Similarly, articles about the politician can be found on many different newspapers and blogs. Also, a Web search result that gets the user close to the result often includes links for browsing to the desired content. Documents in an enterprise context do not include links with which to browse between documents and that can be used to compute relevance. As the above differences imply, the content-based search techniques used on the Web, while helpful, do not fully address the problem of enterprise search and document access, and other techniques to enhance web-type text queries are needed.

In this paper, we present a method for providing search and navigation options to the user through the use of metadata, including computed document genre, and through the use of the document collection file structure. While most documents today are born digital, they often have a period of their life-cycle as paper documents. This happens because they need to be signed, annotated, or handed from one person to another. As a result, a document that starts in an easy-to-read document format (e.g. Microsoft Word) often becomes a scanned document later. The content of the scanned document may be mostly reconstructed through OCR but the result is that all types of documents, whether they start out in a word processor, a spreadsheet, a database, or presentation software, end up as the same document format. Computing document genre helps to provide differentiation of these documents. Another kind of document analysis is used to determine representative keyphrases of documents to provide a quick overview of each document.

Enterprise document collections are often organized in hierarchies similar to directory trees in file systems. These hierarchies are representations of the policies and practices of the

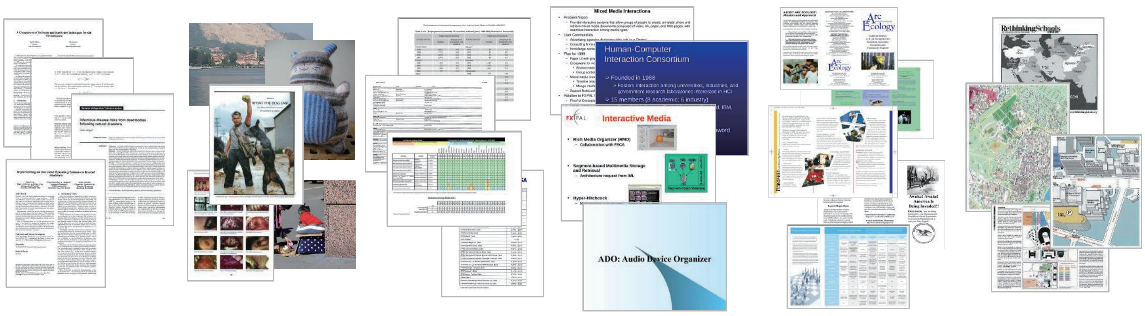

Human-Computer Interaction Consortium
Interactive Media
ADO: Audio Device Organizer
Arc Ecology
RethinkingSchools

Figure 1. Sample document pages from the genres of (a) papers (b) photos (c) tables (d) slides (e) brochures (f) maps.

organization, often partly mapping to the organizational structure and partly to the different roles/activities of people in the organization. While newer interfaces allow for access to documents within multiple categories via tags or other mechanisms, the user experience is still relatively similar: users view the set of categories or directories and navigate through the options displayed to locate documents of interest. To improve system support for such navigation and selection, we combine the data-oriented document analysis with novel interface design to support Web-based repositories for business customers. A facet-based interface designed to run in a Web browser provides a rich user experience enabling a combination of search and navigation-based location strategies. These capabilities are currently part of the new *DocuBrowse* environment.

The next sections present a description of the document analysis component with an emphasis on genre identification and an overview of the mixed-initiative interface for browsing and searching document collections. We conclude with a vision of how such technologies will change the way people and businesses will store and retrieve documents.

2. Genre Identification

Documents are often classified and searched for based on words and topics. However, documents can also be classified by another independent attribute: genre. Example genres in literature include poetry, fiction, and drama. In enterprise search and browsing, a different set of document genres than those used to describe literature are needed.

We have developed the DocuBrowse system for searching and browsing of an FXPAL document repository. In the repository, the documents are created using a variety of tools that produce documents in different formats, such as PDF, DOC, XLS, JPG, PPT, and the DocuWorks portable document format, XDW. In a simplistic approach to genre identification, the different document formats can be thought of as corresponding roughly to different genres, and a document filename extension could be a surrogate for document genre. However, a document creation tool is often used to create documents in more than one genre. For example, in addition to its primary use in creating slides, PowerPoint is also used to create figures and drawings for papers, and also to create hand outs of screen shots and photos. Thus a PowerPoint file may actually not be a slide, but another genre, such as photo.

In addition, some extensions, including PDF and XDW, are associated with many genres, since files created in different formats are often converted to PDF or XDW for its portable representation. Thus, although file extensions can be used as a facet during browsing and search of a document corpus, the ability to accurately estimate the genres characterizing a document for use as search facets can enable more precise search results. For example, if a user were looking for slides from talks, and looked only at PowerPoint files, the user will be presented with extra files that are not slides; furthermore, the user will not see slides that are in other formats, such as PDF or XDW.

In genre identification for DocuBrowse, we categorize the documents into a small number of genres, roughly corresponding to the genres usually associated with document format types: technical paper, slides, table, and photo. Figure 1 shows examples of pages from each of the six genres that our Genre Identification system (DocuGenre), has been trained to identify. Note the variation within a genre. Also note that while the text might not be large enough to read, the genre of each page is readily apparent.

This observation is one motivation for our approach to genre identification. In contrast to topic categorization, which is highly correlated with word usage, genre is primarily indicated through page layout, and depending on the genre, to a lesser extent with word usage. We have focused on an approach based on features that captures the layout of a document page, since our desired genres are generally visually distinctive.

2.1 Related Work on Genre Identification

In a common image-based approach to genre identification, layout analysis is used to label and identify the boundaries of different types of document regions (e.g., text, image, graphics) from which various features are extracted. Shin and Doermann [12] perform document layout analysis, and extract features which are used in a decision tree classifier. Their identified “genres” correspond to page types: cover page, reference, table of contents, and form pages.

Although layout analysis can be successfully performed for limited domains, general layout analysis is still not robust. Two other image-based approaches to genre identification that do not require layout analysis have been developed by Gupta and Sarkar [2] and by Kim and Ross [8]. Gupta and Sarkar identified salient feature points and performed classification based on the points' locations and local image characteristics. However, their genre classifier was tested on discriminating between only two types of genres: journal articles and memos.

Kim and Ross [8] developed an image-based genre classifier for the first page of a document. They divide the page into a uniform grid of 62 by 62 tiles, count the number of non-white pixels in each tile to compute black pixel density, and use the tile densities as classifier features. They compared the performance of the count feature using Naïve Bayes, Random Forest, and SVM classification methods provided by Weka [15]. Their best-performing image-based genre classifier performance was Naïve Bayes, but the performance was relatively poor and is meant to be used in conjunction with a text-based genre classifier.

2.2 The DocuGenre System

In the DocuGenre system, we take an image-based, document-oriented approach that tries to capture layout features without explicitly performing layout analysis. In particular, we tile each page image and extract document-based image features to characterize each tile. Genre identification is performed per *page*, and then *document* genre is estimated based on the genre estimates for the pages in the document.

We developed DocuGenre using a corpus of documents crawled from the web, some of them printed and scanned by us, and some directly converted to JPEG. DocuGenre was trained to identify five genres: maps, technical papers, photos, tables, and fold-up brochures. To evaluate performance, DocuGenre was compared against our implementation of the Kim and Ross system [8], and we observed DocuGenre to have noticeably and statistically significant better performance.

We then adapted DocuGenre to identify four genres for use with DocuBrowse: slides, technical papers, photos, and tables. In the DocuBrowse document repository, the documents are available as image files (JPEG) and the textual content and the locations of the word bounding boxes are available from OCR. We use the bounding box information as well as other image characteristics in computing a feature set as described in the next section.

2.3 Features

We developed a set of features that capture local document characteristics, such as lines of text or text size, *within a tile*. The tiles must be large enough to extract document characteristics within each tile and at the same time small enough so that the different region types (e.g., heading, figure, body text) remain distinct. Empirically, we have found that dividing each page image into a grid of 5 tiles horizontally by 5 tiles vertically, for a total of 25 tiles, meets our requirements.

Because an increasing number of documents are created and printed in color, the features must characterize color documents as well as grayscale and binary image documents. To handle color images, the images are preprocessed using edge detection followed by a morphological dilate. This preprocessing creates a grayscale image where the characters are dark relative to the background. Features are then extracted from the preprocessed color images using the techniques developed for binary and grayscale images.

The following features are computed for each *tile*: (1) *Image density*. This feature is the primary image feature used by Kim and Ross [8]. (2) *Horizontal pixel projection*. The pixels in a tile are projected horizontally and the text rows are identified and statistically characterized in a histogram. If OCR word bounding box locations are available, the "filled" word bounding boxes are projected instead of pixels. (3) *Vertical pixel projection*. Similar to horizontal pixel projection. (4) *Color correlogram*. To reduce the number of correlogram coefficients, feature selection is performed using mRMR (minimum Redundancy Maximum Relevance feature selection) [9] to select a subset of 50 features.

Three *page*-based features are computed: (1) *Horizontal Lines*. To identify rules on a page. The number of lines is noted and the lines lengths are quantized into a histogram. (2) *Vertical Lines*. Vertical line histograms are computed similarly. (3) *Image size*. The width and height of the image.

2.4 Classification

Each document may be tagged with zero or more genres. For example, a document that is a talk containing a lot of slides that are photos may be tagged as both slides and photos. To handle tagging with multiple genres, a separate classifier was trained for independently identifying each genre.

In developing the genre identification classifiers, a corpus of documents composed of 3205 pages from 928 documents was labeled with the targeted genres. We labeled the pages using a tool for selecting pages belonging to a specified genre. Each page was labeled with *0 or more genres*, with the criterion that at least half of the page must be of a particular genre to be labeled that genre. A total of 2073 labels were assigned. For our evaluation studies, the labeled data was partitioned by document into three sets with equal numbers of documents (ignoring round-off) containing pages of a given genre.

Based on the competitive performance of SVMs for many classification tasks, we used an SVM classifier, SVMlight [6]. A separate classifier using a one-against-many model was trained for each genre for each of the three data partitions. One data partition was used for training the SVM, and a second data partition was used to tune each genre model. In tuning a model, parameters are selected to maximize the *F1*-score, the harmonic mean of precision, *P*, and recall, *R*, which is computed as $F1 = 2PR/(P + R)$. The "optimal" model for each genre is used to classify the page images in a third data partition.

2.5 Page Genre Identification Performance

We compared the performance of DocuGenre against the image-based classifier of Kim and Ross [8], which is the closest work to ours in terms of the types of genres identified and the use of an image-based approach. Based on the description in their paper, we implemented their image-based genre classifier as a baseline for comparisons. We ran both the plain and the kernel density estimation versions of the Weka Naïve Bayes classifier referenced in Kim and Ross [8] on our original data set with five labeled classes: brochures, maps, technical papers, photos, and tables.

We also ran the Weka Random Forest classifier on the same image features. For our data set, the overall performance, as measured by mean F1, was better with the Random Forest classifier than either version of Naïve Bayes. We thus used the Kim and Ross features with the Random Forest classifier (KR+RF) as the baseline system.

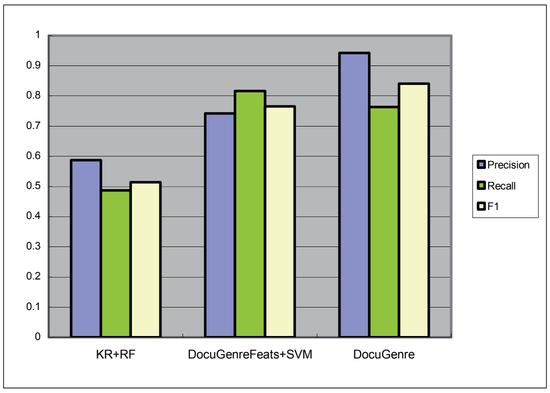


Figure 2. Performance comparison of the baseline Kim & Ross, DocuGenre features + SVM, and DocuGenre

In Figure 2, we refer to our image-based genre identification system as DocuGenre, and refer to the features as DocuGenre features. The mean precision, recall, and F1 are shown for: the baseline system (KR+RF), our DocuGenre features with an SVM classifier (DocuGenreFeats+SVM), and our full DocuGenre system (DocuGenre). Note the improved performance of both DocuGenreFeats+SVM and DocuGenre over the baseline KR+RF system.

2.6 Application to DocuBrowser

Compared to the genres originally used in developing DocuGenre, for the DocuBrowse corpus the "Brochures" and "Maps" categories are not relevant, while a "slides" category needed to be added. To create a classifier that can identify slides, we labeled three sets of 51 PowerPoint files by page for training, tuning, and evaluation. DocuGenre genre models were then trained and tuned using the labeled files. We evaluated DocuGenre on this modified corpus containing additional PowerPoint files using a 3-way jackknife approach. The results for identifying each of the four genres are shown in Figure 3.

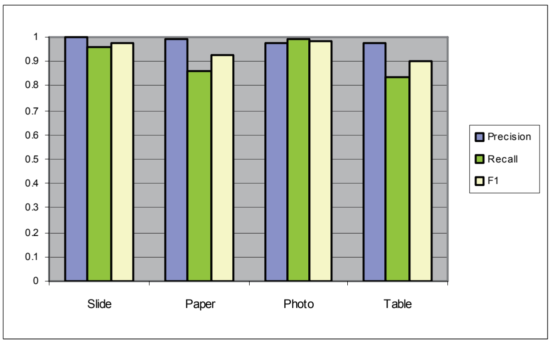


Figure 3. DocuGenre performance in identifying the DocuBrowse genres by page.

2.7 Document Genre Identification

To tag the documents in the DocuBrowse corpus,

$$S_d(g) = \frac{1}{2P}\sum_{p,c}\max(\min(s(p,g,c),1.0),0.0)$$

the four DocuGenre document genre identifiers were used to score each page of all the documents in the corpus. For the DocuBrowse task, the emphasis is on high recall, and our method of combining the page scores for tagging *documents* by genre bears this in mind. A document genre score, $S_d(g)$, for document d being genre g is computed for each genre as the average of the individual page scores for a document. A page score is the SVM score, clipped to a maximum value of 1.0 and a minimum value of 0.0.

where $s(p,g,c)$ is the SVM score for page p being classified as genre g by classifier c and P is the total number of pages in a document. We used two classifiers per genre.

We did not manually label the documents in the DocuBrowse set by genre, and so do not have formal evaluation results, but the examples given in the DocuBrowse interface section give some indication of the performance of genre identification by document.

3. Keyphrase Selection

Keyphrases that give a sense of the content of a document are used in the document summaries presented in DocuBrowse. A small number of keyphrases that can be compactly presented are needed. It was decided that a larger number of shorter keyphrases would be more informative, and so 10 keyphrases that are up to three words long are selected for each document.

There are a number of ways to identify keyphrases (e.g., [13]). A straight-forward method is by tagging the part-of-speech (POS) of the text and then identifying POS tag sequences that correspond to a noun phrase [13]. Another method is to identify sequences of words between "stop words", or non-content words [1]. More recently, supervised methods which learn how to combine different features have been successfully used. One such system is the KEA system [16]. These systems require a training set of documents labeled with keyphrases. Since we do not have a labeled corpus and it would have been time-consuming to manually label a reasonable size corpus for training a system, we took an unsupervised approach.

Our method identifies sequences of words between stop words and other textual cues, such as punctuation, including PowerPoint bullets, and changes in font style and size, as candidate keyphrases. For each document, the candidate keyphrases are scored and the best N keyphrases selected, where N is prespecified and may be dependent on the amount of screen space available to the application.

To select the best keyphrases, we use a weighted combination of features, similar in spirit to a maximum entropy model. The features are text based and include: (1) number of times a term occurs in the document, (2) number of documents in which a term occurs at least once in an English corpus, (3) number of tokens in the keyphrase, and (4) location of first mention of the term in the document, measured as paragraph number.

The weighted combination of features is given by:

$$Score(k_j, d) = \sum_i \lambda_i f_i(k_j, d)$$

where λ_i is the weight given a feature and $f_i(\bullet\ )$ is the value of feature *i* for keyphrase candidate k_j in document *d*. Once each of the keyphrases is scored, they are then ranked against each other and the best keyphrases are selected for each document.

Evaluation of the keyphrase selection has been limited to visual inspection of the keyphrases displayed by DocuBrowse. As with sentence-based summarization tasks, there are generally many more keyphrases that are suitable keyphrases for conveying the gist of a document than can be displayed, and evaluation is a tricky endeavor. For technical papers, the keyphrases displayed by DocuBrowse have been found to be good.

Although we have described keyphrase selection by document, the approach can be applied to other units of text. Our keyphrase selection system has been used to select keyphrases for each *page* of a document. In these cases, the keyphrases are selected within a document, and additional features and methods are used to reduce redundancy in the selected keyphrases. On the other hand, our keyphrase selection system could also be used to select keyphrases for larger units, such as a subdirectory, although the usefulness would depend on the coherence of the documents in each directory.

4. Collection Search, Visualization, and Navigation

The DocuBrowse Interface combines well-known techniques for supporting document access, including browsing the structure of the document collection, searching content, filtering based on metadata, and presenting recommendations based on past user activity.

4.1 Related Work in Faceted Search and Browsing

Facets have been presented and used in search and browsing systems in a variety of ways, and the set of facets supported varies, depending on the contents of the document repository. A study by Wilson and schraefel [14] found that "a balance of exploratory and keyword searches" was performed using the mSpace faceted browser during early use of the system and in later use, indicating that both browsing and search should be supported. This is a feature in our system and also in the faceted search systems that we will contrast with ours.

The mSpace faceted browser lays out categorical facet values in columns which can be moved to indicate the priority of filtering relationships. Our system also provides for user-ordered filtering by facet, but the ordering is determined by the order that a user specifies facet values that are of interest.

Hearst [3] provides recommendations for the design and layout of *hierarchical*, categorical facets in the Flamenco system and also comments on the use of facets in the eBay Express interface. In Flamenco, which was developed on a collection of fine art images, the result set is shown on the right half of the interface, with result items grouped by the most recently selected facet. Microsoft's FacetLens system [9], which was run on the CHI publication repository and also on a database of 32 years of OCED grant data, lays out facets and their attribute values in rectangular regions, with the result set also presented on the right side of the interface. FaThumb [7] was also developed at Microsoft and provides faceted mobile search.

In contrast with these systems, our system relies on the directory hierarchy to provide the grounding context for users, and results are presented by filtering out parts of the directory hierarchy and by highlighting directories to indicate those that are good matches. In addition, the facets in our system are a mix of pre-specified categories and facets with many values, such as a date range or range of file sizes.

These earlier systems were developed for a browsing task with a relatively homogenous collection of item types, while our "enterprise" document collection contains a variety of document types, such as technical papers and slides. The UpLib system [5] is a "personal digital library system" that also contains a variety of document types, such as technical papers and slides, similar to our document corpus. In UpLib, metadata can be stored with each document. Documents are accessed either by viewing all the documents in the repository or by search over the text and metadata, such as "authors" or "keywords" using the Lucene system. The search-based approach is in contrast to the faceted systems which prompt the user to select a facet value, which reduces the cognitive load on the user.

4.2 Browsing the Document Collection

Enterprise workers typically organize their document collections into sub-collections that group documents based on characteristics of their content, generation, or use. In file system-based document stores, the collection structure is the directory hierarchy. For documents with metadata, documents may be placed based on an ontology of the metadata concepts (e.g. MeSHTerms for the National Library of Medicine).

Acknowledging the centrality to browsing through collections, we created DocuBrowse as a web-based interface that is meant to make

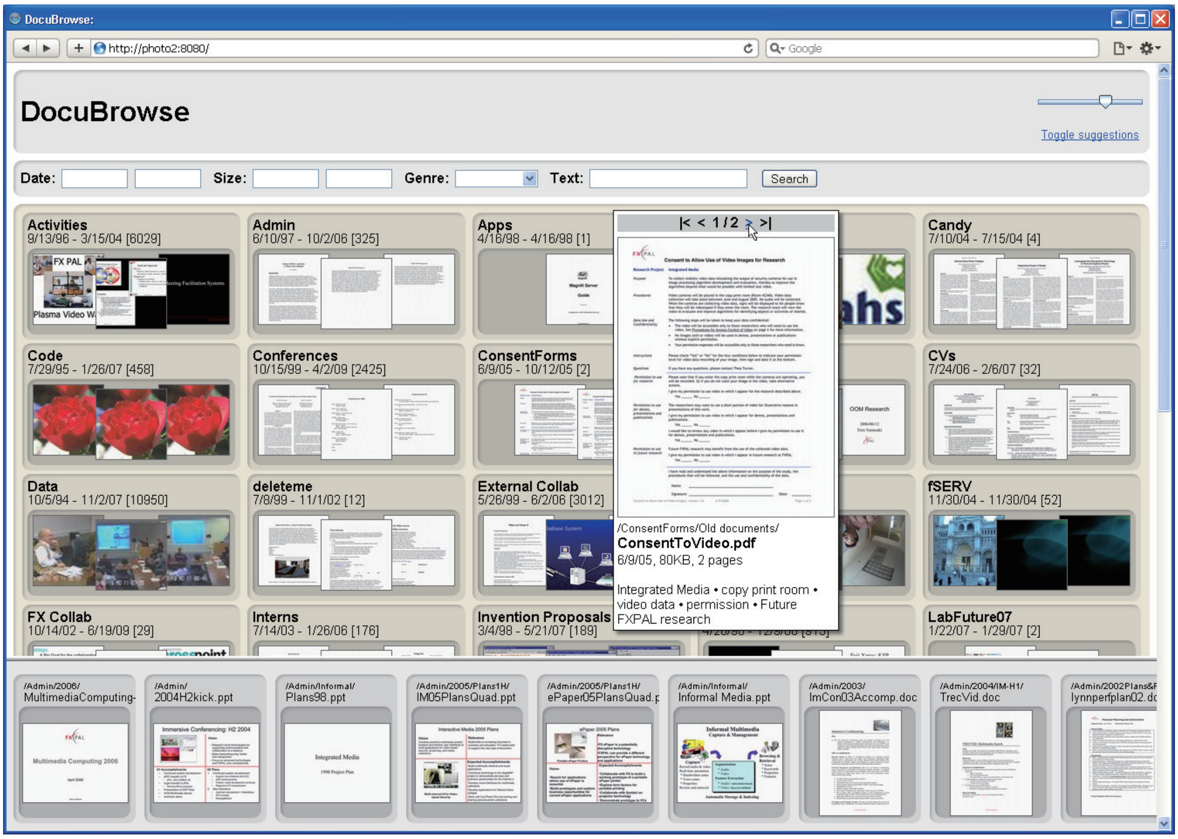


Figure 4. Top-level of the DocuBrowse document hierarchy.

browsing such structures easy and intuitive (see Figure 4). As such, the majority of DocuBrowse's display is used to present the contents of the user's current location in the document collection. Subcollections (e.g. directories) are presented above the individual documents.

Unlike in traditional file systems, DocuBrowse allows users to put a document into more than one "directory." Unlike Windows short cuts, each document entry provides direct access to the original document. In the DocuBrowse prototype, documents are identified by a content hash. This enables automatic duplicate detection so that only one copy of the document has to be stored. As a change to a document changes its identifier, all entries for that document have to be updated. A history mechanism can point to previous versions of a document. To explicitly point to an older version of a document, an entry has to be marked to prevent updates for new document versions.

While the initial document hierarchy is modeled after the file system hierarchy provided by the user, multiple hierarchies can be supported. Each hierarchy is stored in database tables so that no changes to the flat document storage area are required when a hierarchy is added or changed. Other hierarchies may be based on document properties or on external semantic hierarchies such as the Library of Congress classification for books.

4.3 Visualizing Documents

Documents are visualized by a container including the document name, metadata, and automatically selected keyphrases on the left and a thumbnail of the first page of the document on the right. Clicking anywhere in the document opens the document in the document viewer. The document thumbnails and the information text are offered in different sizes that can be changed quickly by using a slider. This slider employs dynamic web technologies to zoom in or out without having to reload the whole page. Thumbnails are created in many different sizes when documents are added to the collection so that thumbnails of a requested size can be shown quickly.

Subcollections, or more simply collections, are visualized by three thumbnails of selected documents in that collection. If fewer than three documents exist in that collection, then only that number of thumbnails is presented. Thumbnails are selected such that a good sample of the documents in the sample is provided. In addition to the thumbnails, the collection's name and statistics about its contents is shown. The statistics include the number of documents and the date range. Clicking on the collection visualization navigates to that subcollection, replacing the current list of collections and documents.

When the mouse lingers over a document or a thumbnail for a document in a subcollection, an interactive tool tip appears that provides easy access to more detailed information about that document. The tool tip also includes a mini document viewer that allows the user to flip through images of the document pages. This lets the user verify quickly if this is the desired document before opening the full-size document viewer.

4.4 Search- and Filter-Based Navigation Support

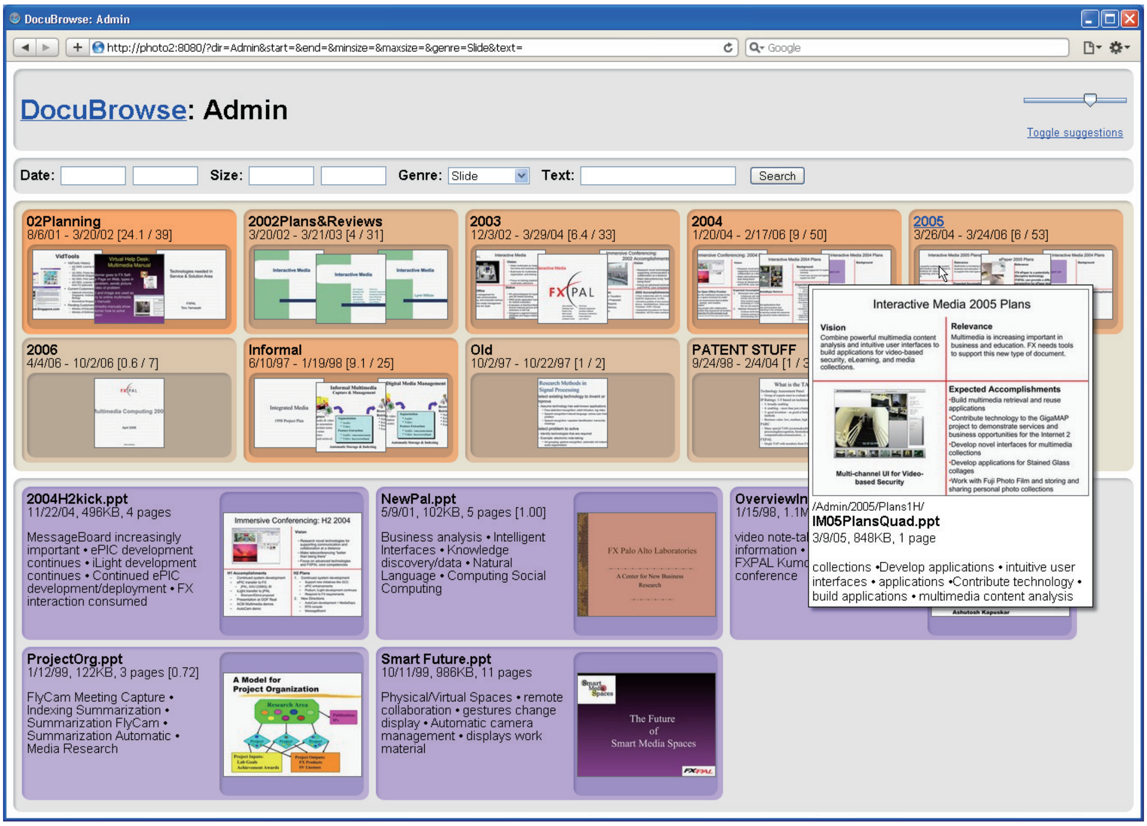
Unlike traditional search systems that display a list of matching documents, DocuBrowse uses a combination of filtering, color-coding, and browsing to present search results. This approach keeps matching documents in context and makes it easy to narrow or widen a search for a subcollection.

To find documents with certain characteristics or contents, DocuBrowse provides filters for different facets of the documents. An important facet is the automatic genre detection. By selecting a genre, only documents matching the genre and directories containing those documents are shown. For fuzzy genre matches, color coding is used to indicate the strength of the match.

In addition to genres, DocuBrowse can filter the results based on document size and date. More options can be provided for document collections where additional metadata facets are available. When the user specifies values for a facet, only those documents that fit those values are shown in the browser. DocuBrowse also supports full-text search, so documents that partially or completely match the terms in the query are displayed in the browser while non-matching documents are filtered out. As with fuzzy genre matches, color coding indicates the quality of the match.

When a search or filter is active, the visualization of subcollection is colored to show where large numbers of matching documents are located. The current visualization indicates where the aggregated document relevance is high. Future visualizations could indicate the best-matching documents (e.g. when the top-ranking documents are the only documents in their directory that match the query).

To illustrate our approach to search, we describe a use scenario. The Director of Research at a research laboratory wants to access details about a presentation system that was created a few years ago. When looking at the top-level view presented by DocuBrowse, she notices the "Admin" directory and realizes that project reviews stored in that directory would be a good source for the desired information. After navigating to that directory, she restricts the view to documents in the "slide" genre (see Figure 5). Among the keyphrases displayed for one of the documents is the word "ePIC" that sounds familiar. To make sure that this is indeed the desired system, she performs a text search with that term. This reduces the view to two subcollections and one document. Clicking on the document displays search clippings that provide sufficient information to identify the system.



DocuBrowse: Admin

http://photo2:8080/?dir=Admin&start=&end=&minsize=&maxsize=&genre=Slide&text=

DocuBrowse: Admin

Toggle suggestions

Date: Size: Genre: Slide Text: Search

02Planning
8/6/01 - 3/20/02 [24.1 / 39]

2002Plans&Reviews
3/20/02 - 3/21/03 [4 / 31]

2003
12/3/02 - 3/29/04 [6.4 / 33]

2004
1/20/04 - 2/17/06 [9 / 50]

2005
3/26/04 - 3/24/06 [6 / 53]

2006
4/4/06 - 10/2/06 [0.6 / 7]

Informal
6/10/97 - 1/19/98 [9.1 / 25]

Old
10/2/97 - 10/22/97 [1 / 2]

PATENT STUFF
9/24/98 - 2/4/04 [1 / 3

2004H2kick.ppt
11/22/04, 496KB, 4 pages

MessageBoard increasingly important • ePIC development continues • iLight development continues • Continued ePIC development/deployment • FX interaction consumed

NewPal.ppt
5/9/01, 102KB, 5 pages [1.00]

Business analysis • Intelligent Interfaces • Knowledge discovery/data • Natural Language • Computing Social Computing

OverviewIn
1/15/98, 1.1M

video note-tal information • FXPAL Kumo conference

ProjectOrg.ppt
1/12/99, 122KB, 3 pages [0.72]

FlyCam Meeting Capture • Indexing Summarization • Summarization FlyCam • Summarization Automatic • Media Research

Smart Future.ppt
10/11/99, 986KB, 11 pages

Physical/Virtual Spaces • remote collaboration • gestures change display • Automatic camera management • displays work material

Interactive Media 2005 Plans

Vision
Combine powerful multimedia content analysis and intuitive user interfaces to build applications for video-based security, eLearning, and media collections.

Relevance
Multimedia is increasing important in business and education. FX needs tools to support this new type of document.

Multi-channel UI for Video-based Security

Expected Accomplishments
•Build multimedia retrieval and reuse applications
•Contribute technology to the GigaMAP project to demonstrate services and business opportunities for the Internet 2
•Develop novel interfaces for multimedia collections
•Develop applications for Stained Glass collages
•Work with Fuji Photo Film and storing and sharing personal photo collections

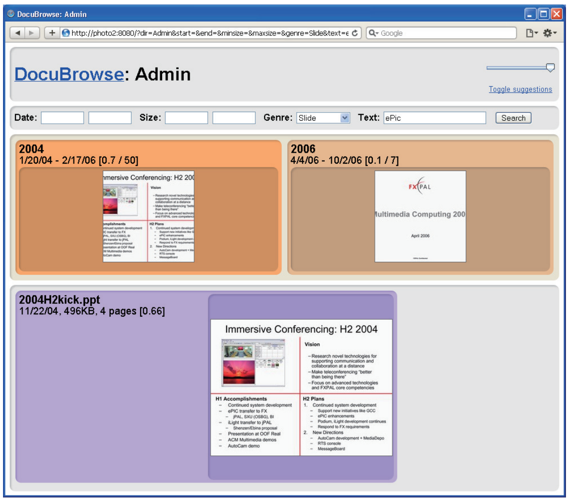
/Admin/2005/Plans1H/
IM05PlansQuad.ppt
3/9/05, 848KB, 1 page

collections •Develop applications • intuitive user interfaces • applications •Contribute technology • build applications • multimedia content analysis

Immersive Conferencing: H2 2004
FX Palo Alto Laboratories
A Center for New Business Research
A Model for Project Organization
The Future of Smart Media Spaces
Ashutosh Kapuskar

Figure 5. Subcollection restricted to documents matching the “slide” genre.

Multiple document facets can be combined to further restrict the matching documents. For example, after having located slides in a document collection, the user can further restrict the matching documents to those that also contain the specified text string (see Figure 6).



DocuBrowse: Admin

http://photo2:8080/?dir=Admin&start=&end=&minsize=&maxsize=&genre=Slide&text=e

DocuBrowse: Admin

Toggle suggestions

Date: Size: Genre: Slide Text: ePic Search

2004
1/20/04 - 2/17/06 [0.7 / 50]

2006
4/4/06 - 10/2/06 [0.1 / 7]

2004H2kick.ppt
11/22/04, 496KB, 4 pages [0.66]

Immersive Conferencing: H2 2004

Vision
– Research novel technologies for supporting communication and collaboration at a distance
– Make teleconferencing "better than being there"
– Focus on advanced technologies and FXPAL core competencies

H1 Accomplishments
– Continued system development
– ePIC transfer to FX
  – jPAL, SXU (OSBG), BI
– iLight transfer to jPAL
  – Shenzen/Ebina proposal
– Presentation at OOF Real
– ACM Multimedia demos
– AutoCam demo

H2 Plans
1. Continued system development
  – Support new initiatives like GCC
  – ePIC enhancements
  – Podium, iLight development continues
  – Respond to FX requirements
2. New Directions
  – AutoCam development + MediaDepo
  – RTS console
  – MessageBoard

Figure 6. Slides containing the text “ePIC.”

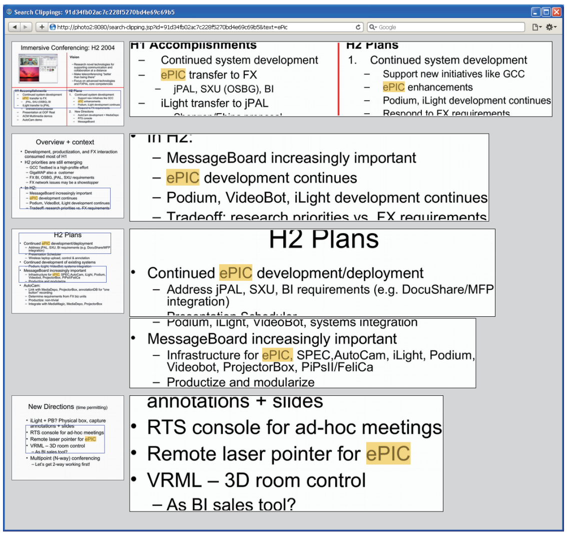
4.5 Recommendations

Unlike in Internet searches, document matches in faceted search are mostly Boolean. Furthermore, there is no simple notion of document importance. We attempt to address this issue by providing document recommendations in the bottom portion of DocuBrowse (see Figure 4). Examples for good recommendations are documents matching the query that other users with similar queries viewed or documents that were recently viewed by other users in the same organization.

The list of recommendations changes as the user interacts with the collection. As our prototype system does not have multiple users or information about the corporate organization, we decided to use a simpler approach for selecting recommendations. Currently, recommendations are provided based on recency and type of access. Documents that have been viewed in the interactive tooltip are available for a short period following their investigation while documents opened in the document viewer appear as suggestions for a longer period of time.

4.6 Document Viewer

The DocuBrowse document viewer provides access to the document pages without requiring other software such as Flash or Adobe Acrobat. It offers two fairly traditional views. One view provides thumbnails of all pages in the documents. Just like in the collection view, a slider allows the user to quickly change the size of the thumbnails. While such a view is available in applications such as Microsoft PowerPoint, it is less common in document viewers. The reading view includes thumbnails on the left with a large view of a single page on the right. It is quite similar to a view provided by Adobe Acrobat.


Search Clippings: 91d34fb02ac7c228f5270bd4e69c69b5
http://photo2:8080/search-clipping.jsp?id=91d34fb02ac7c228f5270bd4e69c69b5&text=ePic
Google
Immersive Conferencing: H2 2004
Overview + context
H2 Plans
New Directions (time permitting)

H1 Accomplishments
– Continued system development
– ePIC transfer to FX
– jPAL, SXU (OSBG), BI
– iLight transfer to jPAL

H2 Plans
1. Continued system development
– Support new initiatives like GCC
– ePIC enhancements
– Podium, iLight development continues
– Respond to FX requirements

In H2:
– MessageBoard increasingly important
– ePIC development continues
– Podium, VideoBot, iLight development continues
– Tradeoff: research priorities vs. FX requirements

H2 Plans
- Continued ePIC development/deployment
  – Address jPAL, SXU, BI requirements (e.g. DocuShare/MFP integration)

– Podium, iLight, VideoBot, systems integration
- MessageBoard increasingly important
  – Infrastructure for ePIC, SPEC,AutoCam, iLight, Podium, Videobot, ProjectorBox, PiPsII/FeliCa
  – Productize and modularize

annotations + slides
- RTS console for ad-hoc meetings
- Remote laser pointer for ePIC
- VRML – 3D room control
  – As BI sales tool?

Figure 7. Clippings of document pages containing the search text.

The snippet view displays more details for full text search results. It shows the thumbnails of pages with matching terms and a larger view of the snippets of text in which those terms appeared (see Figure 7). This view provides a quick view of text matches that are too small to see in a thumbnail view. By showing the outlines of snippets in the page thumbnails, context for the snippets is provided.

5. Conclusions

We presented a new approach for searching and browsing enterprise document collections. To support scanned-in documents and to better classify electronic documents, we utilize an automatic genre identifier that can determine genres such as papers, slides, tables, and photos. We also automatically determine keyphrases that provide users with a quick overview of the document content. These techniques are part of a web-based system for accessing document collections. Unlike traditional document search systems, our system presents search results within the user-created document hierarchy by only showing matching documents and directories and highlighting promising areas. We expect this novel approach to simplify access to enterprise document collections.

6. Acknowledgements

We have been helped by many people in this work. We are grateful to Professor Frank Shipman, Department of Computer Science, Texas A&M University, for major contributions to the conception and design of DocuBrowse. We would like to thank Matt Cooper, Gerry Filby, and Yijuan Lu for their contributions to the extraction of image features, and Yoko Sekita for her work in putting together the scanned document corpus. We also would like to thank Bee Liew for her contributions in porting the keyphrase code to Java and adapting it to work with the DocuBrowse data.

7. Trademarks

- Microsoft, Windows, PowerPoint are trademarks or registered trademarks of Microsoft Corporation.
- Adobe, Acrobat are trademarks or registered trademarks of Adobe Systems Incorporated.
- All brand names and product names are trademarks or registered trademarks of their respective companies.

References

[1] F. Chen, S. Putz, D. Brotsky. Automatic method of selecting multi-word key phrases from a document. US Patent 5745602.

[2] M.D. Gupta and P. Sarkar. A shared parts model for document image recognition. In Proc. of the Ninth International Conference on Document Analysis and Recognition, pp. 1163–1172, 2007.

[3] M. Hearst. Design recommendations for hierarchical faceted search interfaces. Proceedings of the ACM SIGIR Workshop on Faceted Search, 2006.

[4] D. Hawking. Challenges in enterprise search. Proceedings of ADC 2004.

[5] W. Janssen and K. Popat. UpLib: a universal personal digital library system. Proceedings of the 2003 ACM Symposium on Document Engineering, pp. 234-242, 2003.

[6] T. Joachims, Making large-scale SVM learning practical. Advances in Kernel Methods - Support Vector Learning, B. Schölkopf and C. Burges and A. Smola (ed.), MIT-Press, 1999.

[7] A. Karlson, G. Robertson, D. Robbins, M. Czerwinski, and G. Smith. FaThumb: a facet-based interface for mobile search. In Proceedings of CHI'06, pp. 711-720, 2006.

[8] Y. Kim and S. Ross. Examining variations of prominent features in genre classification. In Proc. of the 41st Annual Hawaii International Conference on System Sciences, 2008.

[9] B. Lee, G. Smith, G. Robertson, M. Czerwinski, D. Tan. FacetLens: exposing trends and relationships to support sensemaking within faceted datasets. Proceedings of CHI '09, pp. 1293-1302, 2009.

[10] R. Mukherjee and J. Mao. Enterprise search: Tough stuff. Queue. April 2004, pp. 36-46.

[11] H. Peng, F. Long, and C. Ding. Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. IEEE Transactions on Pattern Analysis and Machine Intelligence, 27:1226–1238, 2005.

[12] C. Shin and D. S. Doermann. Classification of document page images based on visual similarity of layout structures. In Proc. SPIE Conference on Doc. Recog. and Retrieval Ⅶ, 2000.

[13] P. Turney. Extraction of keyphrases from text: evaluation of four algorithms. National Research Council of Canada Technical Report ERB-1051, 1997.

[14] M.L. Wilson and m.c. schraefel. A longitudinal study of exploratory and keyword search. Proceedings of the 8th ACM/IEEE-CS Joint Conference on Digital Libraries, pp. 52-56, 2008.

[15] I. Witten and E. Frank. Data Mining: Practical machine learning tools and techniques, 2nd Edition, Morgan Kaufmann, San Francisco, 2005.

[16] I. Witten, G. Paynter, E. Frank, C. Gutwin, C. Nevill-Manning. KEA: practical automatic keyphrase extraction, Proceedings of the Fourth ACM Conference on Digital Libraries, pp.254-255, August 11-14, 1999.

Author's Introductions

Francine Chen
FX Palo Alto Laboratory
Area of specialty: Electrical Engineering and Computer Science (Ph.D), Multimedia Information Access

Andreas Girgensohn
FX Palo Alto Laboratory
Area of specialty: Computer Science (Ph.D.), Human Computer Interaction, Multimedia, ACM Distinguished Scientist

Lynn Wilcox
FX Palo Alto Laboratory
Area of specialty: Mathematical Sciences (Ph.D), Multimedia