Slide Image Retrieval: A Preliminary Study

Guo Min Liew
Department of Computer Science
School of Computing
National University of Singapore
liewguom@comp.nus.edu.sg

Min-Yen Kan
Department of Computer Science
School of Computing
National University of Singapore
kanmy@comp.nus.edu.sg

ABSTRACT

We consider the task of automatic slide image retrieval, in which slide images are ranked for relevance against a textual query. Our implemented system, SLIDIR, caters specifically for this task using features specifically designed for synthetic images embedded within slide presentation. We show promising results in both the ranking and binary relevance task and analyze the contribution of different features in the task performance.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *retrieval models.*

General Terms

Algorithms, Measurement, Human Factors

Keywords

SLIDIR, presentations, slides, synthetic images, image retrieval

1. INTRODUCTION

Scholarly digital libraries (DLs) contain a wealth of information that requires robust retrieval methods to fully utilize them. Very often, these libraries include mixed media, including presentation decks. However, current retrieval techniques concentrate wholly on single modalities such as text documents, photographs or videos. The inability to retrieve useful mixed media results can hamper search and other information seeking tasks in DLs.

Cutting edge work in computer science and other disciplines is often presented using A/V devices such as presentations to facilitate learning and increase interactivity. Synthetic images such as tables, charts and graphs contained in these presentations complement rather than replace primary information sources such as conference proceedings and textbooks, by providing a bountiful yet alternative avenue for understanding the topic. From pedagogy, we also know that different learners respond differently to different modalities. Slide image retrieval can be seen as a complementary source of information and relevance to standard full-text retrieval. A learner browsing a presentation may be able to glean a summary of the topic or answer doubts or questions on topics unclear in the full text.

In this paper, we make a first attempt to explore the area of slide image retrieval, that is, to retrieve synthetic images (as opposed to natural photographs) in presentations given textual queries. In particular, we focus on graphical slides, defined as slides containing synthetic images, such as charts, tables, flow diagrams and figures. We investigate this problem subclass for two primary reasons: 1) textual slides can already be retrieved by using textual IR approaches, and 2) images present a different modality for relevance judgments.

For example in such a slide image search engine, the query "Hidden Markov Model" or HMM should return a diagram showing an instance of the model topology and not a title slide of a presentation on or featuring the use of HMMs (Figure 1).

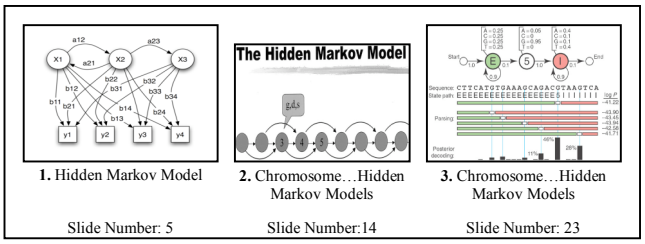


Figure 1. SLIDIR retrieval results for the query "hidden markov model".

2. SLIDE IMAGE RETRIEVAL

Let us state the problem formally:

Given: A corpus of slide presentations S and a query Q.

Output: A ranked list of slide images $I' \in S$, relevant to Q. Note that a slide image may only be a subsection of a slide, when there is more than one image placed together on a slide.

As noted, using text retrieval techniques we can retrieve slides containing text quite readily by treating each slide as a document. However, this technique may not retrieve images that do not contain target text. Content-based Image Retrieval (CBIR) techniques may work better, but require the user to specify a source image as a query. While such content-based systems have been successful in the task similar image search, we cannot expect users to create images as queries.

A blend of these two approaches seems most feasible in which we combine both text and image based features together, as is done in typical image search engines on the web, such as Google Image search.

We report on SLIDIR, a system that retrieves and ranks images extracted from slides given a textual query. SLIDIR differs from general image search engines, as it focuses solely on slide image retrieval from presentation sets. Furthermore, to the best of our knowledge, SLIDIR is the first system specifically designed to retrieve and rank synthetic images. It is architected as a standard, feature-based supervised machine learned system, as shown in the architecture diagram (see Figure 2). However, as the system needs

JCDL'08, June 16–20, 2008, Pittsburgh, Pennsylvania, USA.

to rank images rather than make binary classifications of relevance, we use regression to produce continuous valued output.

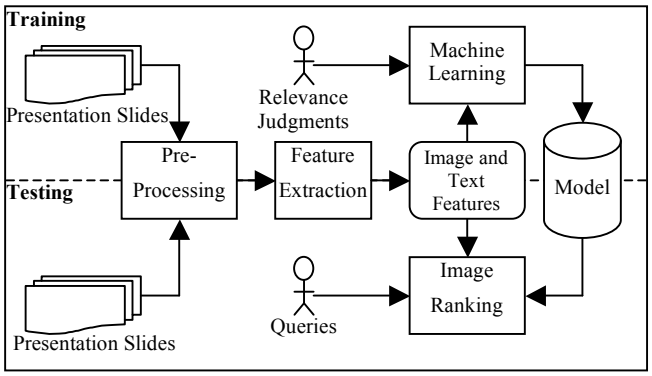


Figure 2. SLIDIR Architecture.

SLIDIR's processing pipeline requires a significant set of preprocessing to extract relevant features from images embedded in the slides. We discuss these preprocessing stages first, before describing how the regression model is built and utilized to rank images against a text query.

2.1 Related Work

Text-based Image Retrieval traditionally involves text-based categorization and classification schemes to describe the image. For example, metadata such as keywords or image filename pertaining to an image are tagged to it for retrieval [2]. One of the advantages of text-based indexing includes the ability to represent both general and specific instantiations of an object at different complexity levels [3]. Automatic image tagging attempts to reduce the workload required for manual tagging by making use of metadata generated automatically using captions and keywords. A form of manual tagging employed in [1] allows end-users to re-label images based on their perceived relevance. As many images lack any surrounding text to be used in tagging, detection and recognition of embedded text in images are also carried out, using techniques such as spatial variance and color segmentation [10]. OCR can also be used for textual content extraction [1].

Content Based Image Retrieval (CBIR) has been generating substantial interest, employing color, texture and shape features. Color refers to the computation of color histograms to obtain the color similarity required for identification [6]. Texture tends to be modeled as a two-dimensional gray level variation to locate visual pattern in images as well as their spatial definition. For example, a texture thesaurus was developed for the matching of the texture areas in an image to words that represent texture attributes [5]. Queries using shape, on the other hand, uses features such as lines, boundaries and aspect ratio. Often, segmentation or edge detection is applied to an image before determining the shapes.

2.2 Preprocessing

Pre-processing in SLIDIR is first carried out to extract visual images in the slides. This consists of background separation and block segmentation executed sequentially.

Background Separation. Unlike natural images and typical textual images, we can take advantage of the fact that presentation slides usually have a consistent background used throughout the presentation set, with few exceptions (e.g., title slides).

Our method thus uses this property to perform *collective* background separation over a set of slides. We poll each pixel over all *n* slides for the RGB color values and assign the dominant (most frequent) RGB value as the background color. Foreground pixels are thus any pixels that differ from the background.

This method works well in presentation sets containing consistent, simple backgrounds where a large number of slides are available for collective separation. Presentation sets containing few slides or complicated backgrounds tend to produce background with substantial noise. However, as our focus is on slide image retrieval, we select presentation slides such that our method works well for the application domain of synthetic images.

Block Segmentation then separates the foreground into individual objects. Existing block segmentation methods are typically iterative. Top-down [8], bottom-up and hybrid approaches have been proposed.

We employ a top-down approach using projection profile cuts similar to [8] due to the regular layout nature of a presentation slide. This algorithm involves first pushing all pixels in each row of the slide to the left before scanning each row to locate vertical gaps wider than a predetermined threshold, T. Horizontal cuts are then made across gaps wider than T. The process is repeated, this time pushing pixels in each column to the bottom and making vertical cuts on horizontal gaps wider than T.

2.3 Features

Once preprocessing of a slide corpus is complete, we have a set of slide images *I* for which we extract representative features from and subsequently index. The focus of our study is on exploring how different textual and image features affect ranking accuracy. A mixture of textual and image features were chosen, resulting in a total of 13 features. The scoring formula of each image feature is given in Table 1.

Our hypothesis was that matching text is a key factor in slide image relevance but that other image features play a role in improving the relevance of search results. To this end we explored text, image and presentation features.

Text Features. We used an in-house PowerPoint extraction program that uses Microsoft's internal API to extract specific fields of text (e.g., slide title, main slide text/bullets) from each slide, generating source strings for the computation of features 2-5. We extract and differentiate these fields as we hypothesize the different text would impact relevance judgments differently. For slide images that are without text, features 1 and 2 yield no text, and the importance of an image needs to be deduced from the textual context of its neighboring slides. Aside from these strings, the presentation title is extracted from the presentation metadata (for feature 6), and all of the text is concatenated together for the baseline system (for feature 1). To extract text that appears in a segmented image (for feature 7; e.g., labels on block diagrams, legends in charts), we convert the slide image to PDF and use the optical character recognition (OCR) functions of Adobe Acrobat Reader[1] to recover the text elements. Such embedded text is important low-level details and words that may indicate the relevance of the image.

Once text strings are extracted, we need to convert them into continuous feature values. For each field string, we use the

[1] http://www.adobe.com

Lucene IR library to compute the similarity between the extracted text and each query Q, resulting in a per-query feature value. Lucene scoring uses a combination of Vector Space Model (VSM) and Boolean model. The extracted text features (features 1-7 and 10) are individually indexed and the corresponding scores for each query are obtained using Lucene APIs.

Table 1. Textual and Image features used in SLIDIR.

No.	Feature	Remarks
Textual Features		
1	All Text	Extract directly from the PPT; use default Lucene similarity to calculate a numeric score
2	Slide Text	
3	Next Slide's Text	
4	Previous Slide's Text	
5	Slide Title	
6	Presentation Title	
7	Slide Image Text	Extract using OCR; again score with default Lucene similarity to query
Image Features		
8	Image Size	Percentage of slide area [0.0-1.0]
9	Number of Colors	Number of distinct colors in image
10	Slide Image Type	Image Classification; use default Lucene similarity to calculate a numeric score
Presentation Features		
11	Slide Order	Relative position of the slide with respect to the presentation [0.0-1.0]
12	Slide Image Position	$\sqrt{((Cxi - Cxs)^2 + (Cyi - Cys)^2)}$ C_{xi}, C_{yi} = center pixel position of image C_{xs}, C_{ys} = center pixel position of slide
13	# Images in Slide	1 / Number of images in the slide image is in

Image Features. We incorporated two basic image features: size of image, number of colors used in the image, which are straightforward to calculate. Image size and position can help indicate the importance of the image relative to the slide's other images and text. We also incorporate a third feature, the image's synthetic image type. Our implementation utilizes the NPIC system [7], whose classification scheme specializes in synthetic images. NPIC's scheme factors synthetic images into a two-level hierarchy, which is detailed enough for our purposes, compared with other image classification work [4] (typically dealing with natural images). For example, the high level category of "figures" is further sub-classed into block diagrams, tables, graphs and pie charts, all of which are used substantially in presentation slides. Such classification may provide a learnable preference between the image classes.

Presentation Features. Finally, we incorporate features of the image relative to the presentation itself. These features included the number of images on the slide, the image's position on the slide (centered; off-centered), as well as the relative position of the slide within all slides in the presentation. These features might help rank similar images within the same presentation set.

3. EVALUATION

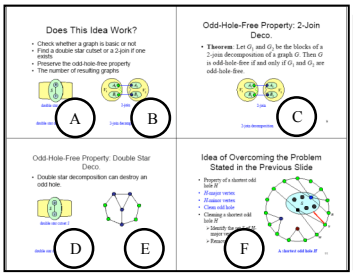
Once the feature vectors of images in the training data have been constructed, a regression model can be built. Given such processed training data and sample training queries, we obtain each image's rank for the queries by asking human subjects. These rankings are then employed to learn weights for each feature using linear regression. We then can employ these learned weights with the features to assess relevance of (new) slide images for unseen queries.

3.1 Data Collection

We collected the per-query image ranking judgments from a set of nine volunteer participants. Each subject was given a number of presentation-query pairs, in which the presentation was already known to be relevant to the query. For every query given, the participants selected and ranked up to five images based on the images' relevance to the query. Each image-query pair thus constitutes a training data point: ranked images were assigned their ranking (1-5) and unranked (irrelevant) images were assigned a default, irrelevant rank of 100.

A total of 120 images were extracted from 4 arbitrarily chosen presentation sets. For each presentation set, the participants were given a slide handout for each presentation (similar to Figure 3(a)) and a list of 10 queries. The participants were tasked to select and rank up to 5 appropriate images for every query. Participants were given 30 minutes to complete the queries. Figure 3(b) shows a sample of the ranking questionnaire given.

For every query, the images that are unranked by a user are given a penalty rank of 100. For example, for the first query in Figure 3, other than the ranked images C, A and E, the unranked images (B, D, F, G, H and I) are all given a rank of 100.



Presentation Set 3							
For each query, choose the most relevant images from the slides and and rank them accordingly. Stop if no more relevant images (see sample)							
No.	Query	Most Relevant Image by Rank					Comments
		1	2	3	4	5	
1	Perfect Graphs	A	C	E	B	na	
2	Odd Hole						
3	2-join						
4	double star						
5	Xinming Liu						

(a) **(b)**

Figure 3. Sample of (a) slide handout and (b) queries with blanks for participant's manual ranking.

Regression Analysis

The feature vectors and survey data are input into a standard machine learning toolkit (Weka [9]). Regression analysis is then carried out with the image rank as the explanatory variable and the image features as the dependent variables. Although relevance is usually considered a binary judgment, we chose to learn a continuous valued function as our main task using regression to simplify the construction of a ranked list.

Different feature configurations of SLIDIR are shown in Table 2. An all text baseline performs quite poorly, resulting in substantial mean absolute error (off by 20 ranks on average). With the incorporation of fielded text features ranking accuracy improves substantially, Further adding image and presentation features improves performance slightly (3% in correlation with human judgments. While this improvement is not particularly substantial, this fits into our hypothesis that other image features can play a role in improving the relevance of search results in additional to matching text. Our current focus in SLIDIR hopes to build on this result and further minimize errors in ranking.

Table 2. Comparison of mean absolute error and correlation.

	All text baseline	Fielded Text	With Presentation and Image Features	Inter-annotator
Mean abs. error	19.36	13.72	13.81	9.40
Correlation	-0.015	0.565	0.581	0.531

Correlation measures the ability of the system to agree in ranking with the aggregate human data collected from our study participants, and is measured on a scale from +1 to -1 (complete agreement to complete disagreement). A correlation level of 0.58 indicates moderate agreement between the system and the human subjects. By decomposing the aggregate data by individual contributor, we can also measure inter-annotator agreement, as shown in the final column of Table 2. Using these figures as a gauge of the upper bound on performance, we see that the full configuration of SLIDIR reduces the ranking error significantly as compared to the baseline, and is comparable in correlation to other human judges.

3.2 Binary Classification

While regression is favored for creating a ranked list, retrieval tasks are more commonly evaluated on binary relevance judgments. We therefore converted our human participants judgments to binary ones (+relevant, -relevant) to assess how a standard classifier would perform given such data. An image is considered relevant as long as it is ranked and not relevant otherwise. Using the SMO support vector machine implementation in Weka and the J48 decision tree classifier, we examine the efficacy of the fully feature set (fielded text + image + presentation features) using standard 10-fold cross validation.

Table 3. 10-fold performance over 8062 instances.

	SVM (SMO)	J48 Decision Tree
Total Accuracy	90.9%	92.5%
Kappa	.54	.59
Relevant Class Prec/Recall/F1	.59/.58/.59	.70/.58/.63

Table 3 reports the evaluation results. General accuracy is quite good in both classifiers, but it is apparent that the precision of the SVM underperforms the decision tree algorithm, indicating that sequential testing is largely adequate for the task.

How important is each of the features used in the classifiers? By comparing the attribute weights generated by the SVM and decision tree classifiers, we can assess how impactful they are. Textual features top the rankings when we analyze the SVM weights: the slide title, embedded text and surrounding text in an image are ranked 1, 2 and 4, respectively. This fits into our expectation that these features when matched indicate a direct reference to the image subject. Surprisingly the matching score of the text in the image itself ranks higher than the text (or bullet points) on the slide; we believe this may be due to cases where very little text is present outside of the slide image. Presentation slide order (feature 11) is ranked third, possibly indicating that presenters may first introduce key ideas towards the beginning of the presentation. Image features were not as significant, with the NPIC image type ranking 9th in importance in the SVM weighting. Contextual features were mixed, with the text of the next slide being considerably more important than text contained on the previous slide.

Table 4. SVM featured in ranked order (normalized attribute weights in parenthesis. -ve values have higher importance).

1. Slide Title (-3.04), 2. Slide Image Text (-0.72), 3. Slide Order (-.41), 4. Slide Text (-0.24), 5. Image Size (-0.09), 6. Next Slide's Text (-0.08), 7. Presentation Title (-0.05), 8. # Images in Slide (-0.0039), 9. Slide Image Type (0.03), 10. Slide Image Position (0.06), 11. Previous Slide's Text (0.36), 12. # of Colors (0.46)

These weights are partially corroborated in the decision tree analysis. Again, textual features were useful features, appearing in the upper portions of the tree. However, differing from the SVM, image features such as number of colors and size are used extensively in the middle levels of the decision tree although slide image type is rarely used. The contextual features are also used less frequently.

4. CONCLUSION

To our knowledge, no existing system explicitly ranks synthetic images from presentation datasets against queries. As graphical images convey complementary information to running text, such systems are of importance to scholarly digital libraries. SLIDIR is an extension of our ongoing work to incorporate slides as first-class objects in scholarly DLs. We show that standard image retrieval techniques work but can be further enhanced using presentation and image specific features.

5. REFERENCES

[1] Denoue, L., Hilbert, D. M., Adcock, J., Billsus, D. and Cooper, M. 2005. ProjectorBox: Seamless presentation capture for classrooms. World Conference on E-Learning in Corporate, Government, Healthcare, & Higher Education.

[2] Fukumoto, T. 2006. An Analysis of Image Retrieval Behavior for Metadata Type Image Database. Information Processing & Management. 42(3), 723-728.

[3] Goodrum, A. A. 2000. Image Information Retrieval: An Overview of Current Research. Special Issue on Information Science Research. 3, 2. 63-66.

[4] Hu, J. and Bagga, A. 2003. Functionality-based web image categorization. In Proc. of WWW (Budapest, Hungary).

[5] Ma, W. and Manjanath, B. 1998. Netra: A toolbox for navigating large image databases. In Proceedings of IEEE. International Conference on Image Processing 1, 568-571.

[6] Stricker, M. and Orengo, M. 1995. Similarity of Color Images. In Proc. of SPIE, 2420, 381-392.

[7] Wang, F. and Kan, M.-Y. 2006. NPIC: Hierarchical Synthetic Image Classification using Image Search and Generic Features. In Proc. of CIVR (Tempe, Arizona, USA), 473-482.

[8] Wang, H., Li, Z. S. and Ragupathi, S. 1997. Document Segmentation and Classification with Top-Down Approach. 1st Int'l Conf. on Knowledge-Based Intelligent Elect. Sys,

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

[10] Zhong, Y., Karu, K. and Jain, A. K. 1995. Locating text in complex color images. In Proc of ICDAR, 146.