Cross-Modal Correlation Mining using Graph Algorithms

Jia-Yu Pan
Carnegie Mellon University
Pittsburgh, U.S.A.
Phone: 412-268-1845
Fax: 412-268-5576
Email: jypan@cs.cmu.edu

Hyung-Jeong Yang
Chonnam National University
Gwangju, South Korea
Phone: +82-62-530-3430
Fax: +82-62-530-3439
Email: hjyang@chonnam.ac.kr

Christos Faloutsos
Carnegie Mellon University
Pittsburgh, PA, 15213, U.S.A.
Phone: 412-268-1457
Fax: 412-268-5576
Email: christos@cs.cmu.edu

Pinar Duygulu
Department of Computer Engineering
Bilkent University
Ankara, Turkey, 06800
Phone: +90-312-290-3143
Fax: +90-312-266-4047
Email: duygulu@cs.bilkent.edu.tr

Cross-Modal Correlation Mining using Graph Algorithms

ABSTRACT

Multimedia objects like video clips or captioned images contain data of various modalities such as image, audio, and transcript text. Correlations across different modalities provide information about the multimedia content, and are useful in applications ranging from summarization to semantic captioning. We propose a graph-based method, *MAGIC*, which represents multimedia data as a graph and can find cross-modal correlations using "random walks with restarts". MAGIC has several desirable properties: (a) it is general and domain-independent; (b) it can detect correlations across any two modalities; (c) it is insensitive to parameter settings; (d) it scales up well for large datasets, (e) it enables novel multimedia applications (e.g., *group captioning*), and (f) it creates opportunity for applying graph algorithms to multimedia problems. When applied to automatic image captioning, MAGIC finds correlations between text and image and achieves a relative improvement of 58% in captioning accuracy as compared to recent machine learning techniques.

Keywords:
Multimedia IS – Annotation
Multimedia IS – Content-based Retrieval
Multimedia IS – Image Retrieval
Multimedia IS – Picture Retrieval

Computer Science – Computer Systems – Multimedia
Computer Science – Computer Systems – Data Resource Management
Computer Science – Computer Systems – Data Mining
Computer Science – Computer Systems – Indexing
Computer Science – Computer Systems – Knowledge Discovery
Computer Science – Computer Systems – Multimedia Database

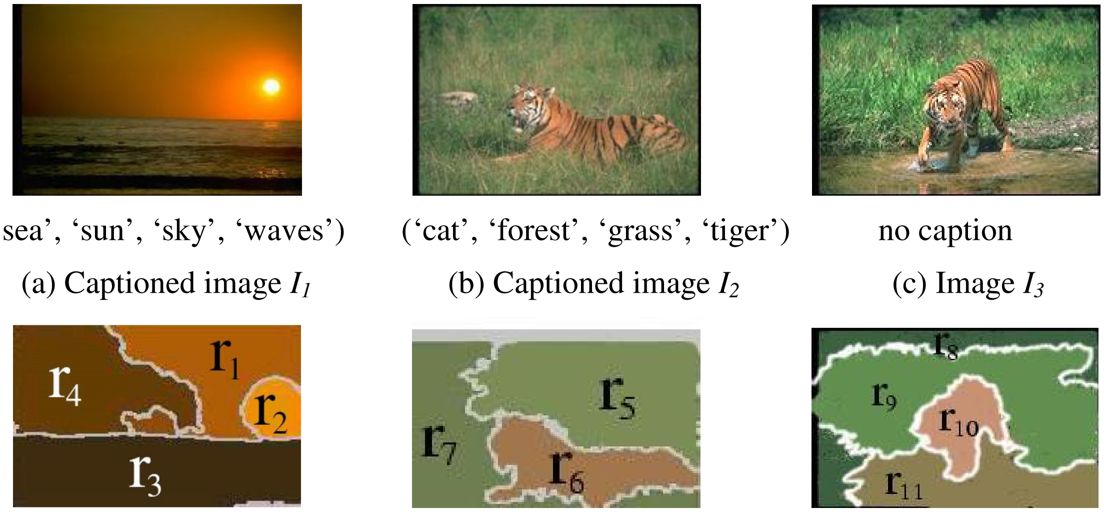
Information System – Knowledge Management IS – Knowledge Discovery
Information System – Knowledge Management IS – Information Search and Retrieval

Library Science – Information Retrieval System

INTRODUCTION

Advances in digital technologies make possible the generation and storage of large amount of multimedia objects such as images and video clips. Multimedia content contains rich information in various modalities such as images, audios, video frames, time series, etc. However, making rich multimedia content accessible and useful is not easy. Advanced tools that find characteristic patterns and correlations among multimedia content are required for the effective usage of multimedia databases.

We call a data object whose content is presented in more than one modality a *mixed media* object. For example, a video clip is a mixed media object with image frames, audios, and other information such as transcript text. Another example is a captioned image such as a news picture with an associated description, or a personal photograph annotated with a few keywords (Figure 1). In this chapter, we would use the terms *medium* (plural form *media*) and *modality* interchangeably.



(‘sea’, ‘sun’, ‘sky’, ‘waves’) (‘cat’, ‘forest’, ‘grass’, ‘tiger’) no caption

(a) Captioned image I_1 (b) Captioned image I_2 (c) Image I_3

(d) (e) (f)

Figure 1: Three sample images: (a),(b) are captioned with terms describing the content; (c) is an image to be captioned. (d)(e)(f) show the regions of images (a)(b)(c), respectively.

It is common to see correlations among attributes of different modalities on a mixed media object. For instance, a news clip usually contains human speech accompanied with images of static scenes, while a commercial has more dynamic scenes with loud background music (Pan and Faloutsos 2002). In image archives, caption keywords are chosen such that they describe objects in the images. Similarly, in digital video libraries and entertainment industry, motion picture directors edit sound effects to match the scenes in video frames.

Cross-modal correlations provide helpful hints on exploiting information from different modalities for tasks such as segmentation (Hsu, Kennedy et al. 2004) and indexing

(Chang, Manmatha et al. 2005). Also, establishing associations between low-level features and attributes that have semantic meanings may shed light on multimedia understanding. For example, in a collection of captioned images, discovering the correlations between images and caption words could be useful for image annotation, content-based image retrieval, and multimedia understanding.

The question that we are interested in is "*Given a collection of mixed media objects, how do we find the correlations across data of various modalities?*" A desirable solution should be able to include all kinds of data modalities, overcome noise in the data, and detect correlations between any subset of modalities available. Moreover, in terms of computation, we would like a method that scales well with respect to the database size and does not require human fine-tuning.

In particular, we want a method that can find correlations among all attributes, rather than just between specific attributes. For example, we want to find not just the image-term correlation between an image and caption terms, but also term-term and image-image correlations, using one single framework. This *any-to-any medium correlation* provides a greater picture of how attributes are correlated, e.g., "which word is usually used for images with blue top," "what words have related semantics," and "what objects often appear together in an image."

We propose a novel, domain-independent framework, *MAGIC*, for cross-modal correlation discovery. MAGIC turns the multimedia problem into a graph problem, by providing an intuitive framework to represent data of various modalities. The proposed graph framework enables the application of graph algorithms to multimedia problems. In particular, MAGIC employs the *random walk with restarts* technique on the graph to discover cross-modal correlations.

In summary, MAGIC has the following advantages:

- It provides a graph-based framework which is domain independent and applicable to mixed media objects which have attributes of various modalities;
- It can detect any-to-any medium correlations;
- It is completely automatic (its few parameters can be automatically preset);
- It can scale up for large collections of objects.

In this study, we evaluate the proposed MAGIC method on the task of *automatic image captioning*. For automatic image captioning, the correlations between image and text are used to predict caption words for an uncaptioned image.

Application 1 (Automatic Image Captioning) *Given a set $\boldsymbol{I_{core}}$ of color images, each with caption words, find the best q (say, q=5) caption words for an uncaptioned image I_{new}.*

The proposed method can also be easily extended for various related applications such as captioning images in groups, or retrieving relevant video shots and transcript words.

In the following of this chapter, we will first discuss pervious attempts on multimedia cross-modal correlation discovery. Then, the proposed method, MAGIC, is introduced. We will show that MAGIC achieves a better performance than recent machine learning methods on automatic image captioning (a 58% improvement on captioning accuracy). Several system issues are also discussed and we show that MAGIC is insensitive to parameter settings and is robust to variations in the graph.

RELATED WORK

Multimedia knowledge representation and application have attracted much research attention recently. Mixed media objects provide opportunities for finding correlations between low-level and concept-level features, and multi-modal correlations have been shown useful for applications such as retrieval, segmentation, classification, and pattern discovery. In this section, we survey previous work on cross-modal correlation modeling. We also discuss previous work on image captioning, which is the application domain on which we evaluate our proposed model.

Multimedia Cross-Modal Correlation

Combining multimedia correlations in applications leverages all available information and has led to improved performances in segmentation (Hsu, Kennedy et al. 2004), classification (Lin and Hauptmann 2002; Vries, Westerveld et al. 2004), retrieval (Wang, Ma et al. 2004; Wu, Chang et al. 2004; Zhang, Zhang et al. 2004), and topic detection (Duygulu, Pan et al. 2004; Xie, Kennedy et al. 2005). One crucial step of fusing multi-modal correlations into applications is to detect and model the correlations among different data modalities.

We categorize previous methods for multimedia correlation modeling into two categories: *model-driven* approaches and *data-driven* approaches. A model-driven method usually assumes a certain type of data correlations and focuses on fitting this particular correlation model to the given data. A data-driven method makes no assumption on the data correlations and finds correlations using solely the relationship (e.g., similarity) between data objects.

The model assumed by a model-driven method is usually hand-designed, based on the available knowledge to the domain. Model-driven methods provide a good way to incorporate prior knowledge into the correlation discovery process. However, the quality of the extracted correlations depends on the correctness of the assumed model. On the other hand, the performance of a data-driven method is less dependent on the available domain knowledge, but the ability to incorporating prior knowledge to guide the discovery process is more limited.

Previous model-driven methods have proposed a variety of models to extract correlations from multi-modal data. Linear models (Srihari, Rao et al. 2000; Li, Dimitrova et al. 2003) assume that data variables have linear correlations. Linear models are computationally friendly, but may not approximate real data correlations well. More complex statistical models have also been used: for example, the mixture of Gaussians (Vries, Westerveld et al. 2004), the maximum-entropy model (Hsu, Kennedy et al. 2004), or the hidden Markov model (Xie, Kennedy et al. 2005). Graphical models (Naphade, Kozintsev et al. 2001; Benitez and Chang 2002; Jeon, Lavrenko et al. 2003; Feng, Manmatha et al. 2004) have attracted much attention for its ability to incorporate domain knowledge into data modeling. However, the quality of the graphical model depends on the correctness of the embedded generative process, and sometimes the training of a complex graphical model can be computationally intractable.

Classifier-based models are suitable for fusing multi-modal information when the application is data classification. Classifiers are useful in capturing discriminative patterns between different data modalities. To identify multi-modal patterns for data classification, one can use either a multi-modal classifier which takes a multi-modal input, or a meta-classifier (Lin and Hauptmann 2002; Wu, Chang et al. 2004) which takes as input the outputs of multiple uni-modal classifiers.

Unlike a model-driven method that fits a pre-specified correlation model to the given data set, a data-driven method finds cross-modal correlations solely based on the similarity relationship between data objects in the set. A natural way to present the similarity relationship between multimedia data objects is using a graph representation, where nodes symbolize objects, and edges (with weights) indicate the similarity between objects.

According to the application domains, different graph-based algorithms have been proposed to find data correlations from a graph representation of a data set. For example, "spectral clustering" has been proposed for clustering data from different video sources (Zhang, Lin et al. 2004), as well as for grouping relevant data of different modalities (Duygulu, Pan et al. 2004). Link analysis techniques have been used for deriving a multi-modal (image and text) similarity function for web image retrieval (Wang, Ma et al. 2004). For these methods, graph nodes are used to represent multimedia objects, and the focus is on finding correlations between data objects. This *object-level graph representation* requires good similarity function between objects (for constructing graph edges) difficult, which is especially hard to obtain for complex multimedia objects.

In this chapter, we introduce MAGIC, a proposed data-driven method for finding cross-modal correlations in general multimedia settings. MAGIC uses a graph to represent the relations between objects and low-level attribute domains. By relating multimedia objects via the constituent single-modal domain tokens, MAGIC does not require object-level similarity functions, which are hard to obtain. Moreover, MAGIC does not need a training phase and is insensitive to parameter settings. Our experiments show that MAGIC can find correlations among all kinds of data modalities, and achieves good performance in real world multimedia applications such as image captioning.

Image Captioning

Although a picture is worth a thousand words, extracting the abundant information from an image is not an easy task. Computational techniques are able to derive low-to-mid level features (e.g., texture and shape) from pixel information, however, the gap still exists between mid-level features and concepts used in human reasoning (Zhao and Grosky 2001; Sebe, Lew et al. 2003; Zhang, Zhang et al. 2004). One consequence of this semantic gap in image retrieval is that the user's need is not properly matched by the retrieved images, and may be part of the reason that practical image retrieval is yet to be popular.

Automatic image captioning, whose goal is to predict caption words to describe image content, is one research direction to bridge the gap between concepts and low-level features. Previous work on image captioning employs various approaches such as linear models (Mori, Takahashi et al. 1999; Pan, Yang et al. 2004), classifiers (Maron and Ratan 1998), language models (Duygulu, Barnard et al. 2002; Jeon, Lavrenko et al. 2003; Virga and Duygulu 2005), graphical models (Barnard, Duygulu et al. 2003; Blei and Jordan 2003), statistical models (Li and Wang 2003; Feng, Manmatha et al. 2004; Jin, Chai et al. 2004). Interactive frameworks with user involvement have also been proposed (Liu, Dumais et al. 2001).

Most previous approaches derive features from image regions (regular grids or blobs), and construct a model between images and words based on a reference captioned image set. Human annotators caption the reference images. However, we have no information about the association between individual regions and caption words. Some approaches attempt to explicitly infer the correlations between regions and words (Duygulu, Barnard et al. 2002), with enhancements that take into consideration interactions between neighboring regions in an image (Li and Wang 2003). Alternatively, there are methods which model the collective correlations between regions and words of an image (Pan, Yang et al. 2004a; Pan, Yang et al. 2004b).

Comparing the performance of different approaches is not easy. Although several benchmark data sets are available, not every previous work reports results on the same subset of images. Various metrics, such as accuracy, mean average precision, and term precision and recall, have been used by previous work to measure the performance. Since the perception of an image is subjective, some work also reports user evaluation of the captioning result. In this chapter, the proposed method is evaluated by its performance on image captioning, where the experiments are performed on the same data set and evaluated using the same performance metric as previous work for fair comparison.

PROPOSED GRAPH-BASED CORRELATION DETECTION MODEL

Our proposed method for mixed media correlation discovery, MAGIC, provides a graph-based representation for multimedia objects with data attributes of various modalities. A technique for finding any-to-any medium correlation, which is based on random walks on the graph, is also proposed. In this section, we explain how to generate the graph representation and how to detect cross-modal correlations using the graph.

Graph Representation for Multimedia Data

In relational database management systems, a multimedia object is usually represented as a vector of *m* features/attributes (Faloutsos 1996). The attributes must be *atomic* (i.e., taking single values) like "size" or "the amount of red color" of an image. However, for mixed media data sets, the attributes can be *set-valued*, such as the caption of an image (a set of words) or the set of regions of an image.

Finding correlations among set-valued attributes is not easy. Elements in a set-valued attribute could be noisy or missing altogether, for example, regions may not be perfectly segmented from an image (noisy regions), and the image caption may be incomplete, leaving out some aspects of the content (noisy captions). Set-valued attributes of an object may have different numbers of elements, and the correspondence between set elements of different attributes is not known. For instance, a captioned image may have unequal numbers of caption words and regions, where a word may describe multiple regions and a region may be described by zero or more than one word. The detailed correspondence between regions and caption words is usually not given by human annotators.

We assume that the elements of a set-valued attribute are tokens drawn from a *domain*. We propose to gear our method toward set-valued attributes, because they include atomic attributes as a special case and also smoothly handle the case of missing values (null set).

Definition 1 (Domain and Domain Token) *The **domain** D_i of (set-valued) attribute A_i is a collection of atomic values, which we called **domain tokens**, which are the values that attribute A_i can take.*

A domain can consist of categorical values, numerical values, or numerical vectors. For example, a captioned image has *m*=2 attributes: The first attribute, "caption", has a set of categorical values (English terms) as its domain; the second attribute, "regions", is a set of image regions, each of which is represented by a *p*-dimensional vector of *p* features derived from the region (e.g., color histogram with *p* colors). As we will describe in the following experimental result section, we extract *p*=30 features from each region. To establish the relation between domain tokens, we assume that we have a similarity function for each domain. Domain tokens are usually simpler than mixed media objects, and therefore, it is easier to define similarity functions on domain tokens than on mixed media objects. For example, for the attribute "caption", the similarity function could be 1 if the two tokens are identical, and 0 if they are not. As for image regions, the similarity function could be the Euclidean distance between the p-dimensional feature vectors of two regions.

Assumption 1 *For each domain D_i (i=1, …, m), we are given a similarity function $Sim_i(*,*)$ which assigns a score to a pair of domain tokens.*

Perhaps surprisingly, with Definition 1 and Assumption 1, we can encompass all the applications mentioned in the introduction. The main idea is to represent all objects and their attributes (domain tokens) as nodes of a *graph*. For multimedia objects with *m* attributes, we obtain a (*m+1*)-layer graph. There are *m* types of nodes (one for each attribute), and one more type of nodes for the objects. We call this graph a MAGIC graph (G_{magic}). We put an edge between every object-node and its corresponding attribute-value nodes. We call these edges *object-attribute-value links* (OAV-links).

Furthermore, we consider that two objects are similar if they have similar attribute values. For example, two images are similar if they contain similar regions. To incorporate such information into the graph, our approach is to add edges to connect pairs of domain tokens (attribute values) that are similar, according to the given similarity function (Assumption 1). We call edges that connect nodes of similar domain tokens *nearest-neighbor links* (NN-links).

We need to decide on a threshold for "closeness" when adding NN-links. There are many ways to do this, but we decide to make the threshold adaptive: each domain token is connected to its *k* nearest neighbors. Computing nearest neighbors can be done efficiently, because we already have the similarity function $Sim_i(*,*)$ for each domain D_i (Assumption 1). In the following section, we will discuss the choice of *k,* as well as the sensitivity of our results to *k*.

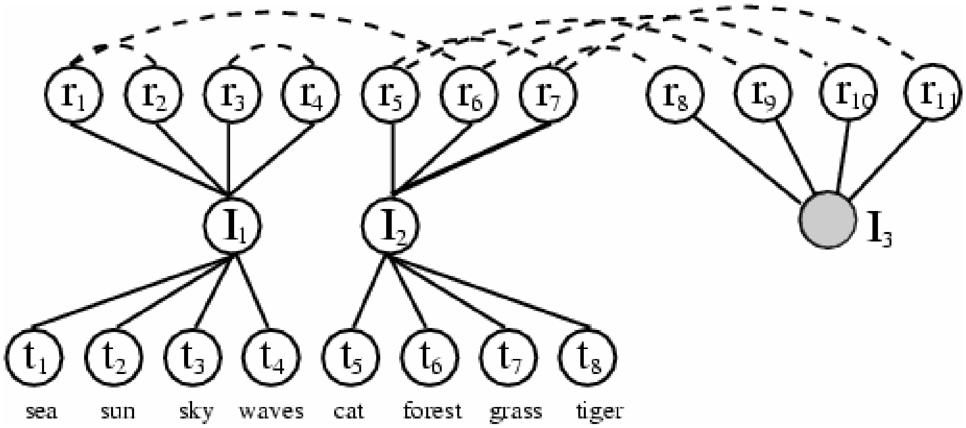


Figure 2. The MAGIC graph (G_{magic}) corresponds to the 3 images in Figure 1. Solid edges: OAV-links; dash edges: NN-links.

We illustrate the construction of the MAGIC graph by the following example.

Example 1 *For the three images {I_1, I_2, I_3} in Figure 1, the corresponding MAGIC graph (G_{magic}) is shown in Figure 2. The graph has three types of nodes: one for the image objects I_j's (j=1,2,3); one for the regions r_j's (j=1,...,11), and one for the terms {t_1,...,t_8} = {sea, sun, sky, waves, cat, forest, grass, tiger}. Solid arcs are the object-attribute-value links (OAV-links). Dashed arcs are the nearest-neighbor links (NN-links), based on some assumed similarity function between regions. There is no NN-link* *between term-nodes, due to the definition of its similarity function: 1, if the two terms are the same; or 0 otherwise.*

In Example 1, we consider only k=1 nearest neighbor, to avoid cluttering the diagram. Because the nearest neighbor relationship is not symmetric and because we treat the NN-links as un-directional, some nodes are attached to more than one link. For example, node r_1 has two NN-links attached: r_2's nearest neighbor is r_1, but r_1's nearest neighbor is r_6. Figure 3 shows the algorithm for constructing a MAGIC graph.

```
Input:
  1. $\boldsymbol{O}$: a set of $n$ objects (objects are numbered from $1$ to $n$).
  2. $\boldsymbol{D_1}, \ldots, \boldsymbol{D_m}$: the domains of the $m$ attributes of the objects in $\boldsymbol{O}$.
  3. $Sim_1(*,*), \ldots, Sim_m(*,*)$: the similarity functions of domains $\boldsymbol{D_1}, \ldots, \boldsymbol{D_m}$,
respectively.
  4. $k$: the number of neighbors a domain token connects to.
Output:
   $G_{magic}$: a MAGIC graph with a $(m+1)$-layer structure.
Steps:
  1. Create $n$ nodes, one for each object. These "object nodes" form the first layer
     of nodes.
  2. For each domain $\boldsymbol{D_i}$, for $i$=1,…,$m$.
     (2.1) Let $n_i$ be the number of tokens in the domain $\boldsymbol{D_i}$.
     (2.2) Create $n_i$ nodes (the token nodes), one for each domain tokens in $\boldsymbol{D_i}$.
           This is the $(i+1)$-th layer of nodes.
     (2.3) Construct the OAV-links from the object nodes to the token nodes.
     (2.4) Construct the NN-links between the token nodes.
  3. Output the final $(m+1)$-layer graph.  The final graph has $N$ nodes, where
                       $N = n + \sum_{i=1,\ldots,m} n_i$ .
```

Figure 3: (Algorithm) $\boldsymbol{G_{magic}}$ **=buildgraph(*O*, {*D*$_1$,…, *D*$_m$}, {*Sim*$_1$,…,*Sim*$_m$}, *k*)**

We use image captioning only as an illustration: the same graph framework can be generally used for other multimedia problems. For automatic image captioning, we also need to develop a method to find good caption words - words that correlate with an image, based on information in the G_{magic} graph. For example, to caption the image I_3 in Figure 2, we need to estimate the correlation degree of each term-nodes (t_1, …, t_8) to node I_3, and the terms that are highly correlated with image I_3 will be predicted as its caption words. The proposed method for finding correlated nodes in the G_{magic} graph is described in the next section.

Table 1 summarizes the symbols we used in the paper.

Symbol	Description
N	The number of objects in a mixed media data set.
M	The number of attributes (domains).
N	The number of nodes in G_{magic}.
E	The number of edges in the graph G_{magic}.
K	Domain neighborhood size: the number of nearest neighbors that a domain token is connected to.
C	The restart probability of RWR (random walk with restarts, RWR).
D_i	The domain of the *i*-th attribute.
$Sim_i(*,*)$	The similarity function of the *i*-th domain.
Image captioning	
$\boldsymbol{I}_{\boldsymbol{core}}$	The given captioned image set (the core image set).
$\boldsymbol{I}_{\boldsymbol{test}}$	The set of to-be-captioned (test) images.
I_{new}	An image in $\boldsymbol{I}_{\boldsymbol{test}}$.
G_{core}	The subgraph of G_{magic}, which contains all images in $\boldsymbol{I}_{\boldsymbol{core}}$.
G_{aug}	The augmentation to G_{core}, which contains information of image I_{test}.
$\boldsymbol{GW}$	The gateway nodes: the set of nodes of G_{core} that are adjacent to G_{aug}.
Random walk with restarts (RWR)	
$\mathbf{A}$	The (column-normalized) adjacency matrix. The (i,j)-element of $\mathbf{A}$ is $A_{i,j}$.
$\mathbf{v_R}$	The restart vector of the set of query objects $\boldsymbol{R}$, where components correspond to query objects have value $1/\lvert\boldsymbol{R}\rvert$, while others have value 0).
$\mathbf{u_R}$	The RWR scores of all nodes with respect to the set of query objects $\boldsymbol{R}$.
$\mathbf{v_q}$, $\mathbf{u_q}$	The $\mathbf{v_R}$ and $\mathbf{u_R}$ for the singleton query set $\boldsymbol{R}=\{q\}$.

Table 1: Summary of symbols used in the paper

Correlation Detection with Random Walks

Our main contribution is to turn the cross-modal correlation discovery problem into a graph problem. The previous section describes the first step of our proposed method: representing set-valued mixed media objects in a graph G_{magic}. Given such a graph with mixed media information, *how do we detect the cross-modal correlations in the graph?*

We define that a node A of G_{magic} is correlated to another node B if A has an “affinity” for B. There are many approaches for ranking all nodes in a graph by their “affinity” for a reference node, for example, electricity-based approaches (Doyle and Snell 1984; Palmer and Faloutsos 2003), random walks (PageRank, topic-sensitive PageRank) (Brin and Page 1998; Haveliwala 2002; Haveliwala, Kamvar et al. 2003), hubs and authorities (Kleinberg 1998), elastic springs (Lovász 1996), and so on. Among them, we propose to use *random walk with restarts* (RWR) for estimating the affinity of node B with respect to node A. However, the specific choice of method is orthogonal to our framework.

The “random walk with restarts” operates as follows: To compute the affinity $u_{\mathrm{A}}(B)$ of node B for node A, consider a random walker that starts from node A. The random walker chooses randomly among the available edges every time, except that, before he makes a choice, he goes back to node *A* (restart) with probability *c*. Let $u_A(B)$ denote the steady state probability that our random walker will find himself at node *B*. Then, $u_A(B)$ is what we want, the affinity of *B* with respect to *A*. We also call $u_A(B)$ the *RWR score* of *B* with respect to *A*. The algorithm of computing RWR scores of all nodes with respect to a subset of nodes ***R*** is given in Figure 4.

Definition 2 (RWR Score) *The RWR score, $u_A(B)$, of node B with respect to node A is the steady state probability of node B, when we do the random walk with restarts from A, as defined above.*

```
Input:
  1. $G_{magic}$: a $G_{magic}$ graph with $N$ nodes (nodes are numbered from $1$ to $N$).
  2. $\boldsymbol{R}$: the set of restart nodes. (Let $|\boldsymbol{R}|$ be the size of $\boldsymbol{R}$.)
  3. $c$: the restart probability.
Output:
  $\mathbf{u_R}$: a $N$-by-$1$ vector of the RWR scores of all $N$ nodes, with respect to $\boldsymbol{R}$ .
Steps:
  1. Let $\mathbf{A}$ be the adjacency matrix of $G_{magic}$. Normalize the columns of $\mathbf{A}$ and make each column sum up to 1.
  2. $\mathbf{v_R}$ is the $N$-by-1 restart vector, whose $i$-th element $\mathbf{v_R}(i)$ is $1/|\boldsymbol{R}|$, if node $i$ is in $\boldsymbol{R}$; otherwise, $\mathbf{v_R}(i)=0$.
  3. Initialize $\mathbf{u_R} = \mathbf{v_R}$.
  4. While($\mathbf{u_R}$ has not converged)
     4.1 Update $\mathbf{u_R}$ by $\mathbf{u_R} = (1\text{-}c)\mathbf{A}\,\mathbf{u_R} + c\,\mathbf{v_R}$ .
  5. Return the converged $\mathbf{u_R}$.
```

Figure 4: $\mathbf{u_R} = \mathbf{RWR}(\boldsymbol{G_{magic}}, \boldsymbol{R}, c)$

Let **A** be the adjacency matrix of the given graph G_{magic}, and let $A_{i,j}$ be the (i,j)-element of **A**. In our experiments, $A_{i,j}$=1 if there is an edge between nodes i and j, and $A_{i,j}$=0 otherwise. To perform RWR, columns of the matrix **A** are normalized such that elements of each column sum up to 1. Let $\mathbf{u_q}$ be a vector of RWR scores of all *N* nodes with respect to a restart node *q*, and $\mathbf{v_q}$ be the "restart vector", which has all *N* elements zero, except for the entry that corresponds to node *q*, which is set to 1. We can now formalize the definition of the RWR score as follows:

Definition 3 (RWR Score Computation) The *N*-by-*1* steady state probability vector $\mathbf{u_q}$, which contains the RWR scores of all nodes with respect to node *q*, satisfies the following equation:

$$\mathbf{u_q} = (1\text{-c})\mathbf{A}\,\mathbf{u_q} + \text{c}\,\mathbf{v_q},$$

where *c* is the restart probability of the RWR from node *q*.

The computation of RWR scores can be done efficiently by matrix multiplication (Step 4.1 in Figure 4). The computational cost scales linearly with the number of elements in the matrix **A**, i.e., the number of graph edges determined by the given database. We keep track of the L_1 distance between the current estimate of $\mathbf{u_q}$ and the previous estimate, and we consider the estimation of $\mathbf{u_q}$ *converges* when this L_1 distance is less than 10^{-9}. In our experiments, the computation of RWR scores converges after a few (~10) iterations (Step 4 in Figure 4) and takes less than 5 seconds. Therefore, the computation of RWR scales well with the database size. Moreover, MAGIC is modular and can continue improve its performance by including the best module (Kamvar, Haveliwala et al. 2003; Kamvar, Haveliwala et al. 2003a) for fast RWR computation.

The RWR scores specify the correlations across different media and could be useful in many multimedia applications. For example, to predict the caption words for image I_3 in Figure 1, we can compute the RWR scores $\mathbf{u_{I3}}$ of all nodes and report the top few (say, 5) term-nodes as caption words for image I_3. Effectively, MAGIC exploits the correlations across images, regions and terms to caption a new image.

The RWR scores also enable MAGIC to detect any-to-any medium correlation. In our running example of image captioning, an image is captioned with the term nodes of highest RWR scores. In addition, since all nodes have their RWR scores, other nodes, say image nodes, can also be ranked and sorted, for finding images that are most related to image I_3. Similarly, we can find the most relevant regions to image I_3. In short, we can restart from any subset of nodes, say term nodes, and derive term-to-term, term-to-image, or term-to-*any* correlations. We will discuss more on this in the experimental result section. Figure 5 shows the overall procedure of using MAGIC for correlation detection.

Step 1: Identify the objects $\boldsymbol{O}$ and the m attribute domains $\boldsymbol{D}_i$, $i=1,\ldots,m$.
Step 2: Identify the similarity functions $Sim_i(*,*)$ of each domain.
Step 3: Determine k: the neighborhood size of the domain tokens. (Default value $k=3$.)
Step 4: Build the MAGIC graph,
  G_{magic} = buildgraph($\boldsymbol{O}$, $\{\boldsymbol{D}_1, \ldots, \boldsymbol{D}_m\}$, $\{Sim_1(*,*), \ldots, Sim_m(*,*)\}$, k).
Step 5: Given a query node $\boldsymbol{R}=\{q\}$ (q could be an object or a domain token),
  (Step 5.1) Determine the restart probability c. (Default value $c=0.65$.)
  (Step 5.2) Compute the RWR scores: $\mathbf{u_R}$ = RWR(G_{magic}, $\boldsymbol{R}$, c).
Step 6: Objects and domain tokens with high RWR scores are considered correlated to q.

Figure 5: Steps for correlation discovery using MAGIC. Functions “buildgraph()” and “RWR()” are given in Figure 3 and Figure 4, respectively.

APPLICATION: AUTOMATIC IMAGE CAPTIONING

Cross-modal correlations are useful for many multimedia applications. In this section, we present results of applying the proposed MAGIC method to automatic image captioning. Intuitively, the cross-modal correlations discovered by MAGIC are used in the way that an image is captioned automatically with words that correlate with the image content.

We evaluate the quality of the cross-modal correlations identified by MAGIC in terms of captioning accuracy. We show experimental results to address the following questions:

- Quality: Does MAGIC predict the correct caption terms?
- Generality: Beside the image-to-term correlation for captioning, can MAGIC capture any-to-any medium correlation?

Our results show that MAGIC successfully exploits the image-to-term correlation to caption test images. Moreover, MAGIC is flexible and can caption multiple images as a group. We call this operation "*group captioning*" and present some qualitative results. For detecting any-to-any medium correlations, we show that MAGIC can also capture same-modal correlations such as the term-term correlations: i.e., "given a term such as 'sky', find other terms that are likely to correspond to it." Potentially, MAGIC is also capable of detecting other correlations such as the reverse captioning problem: i.e., "given a term such as 'sky', find the regions that are likely to correspond to it." In general, MAGIC can capture any-to-any medium correlations.

Data Set and the MAGIC Graph Construction

Given a collection of captioned images $\boldsymbol{I_{core}}$*, how do we select caption words for an uncaptioned image* I_{new}*?* For automatic image captioning, we propose to caption I_{new} using the correlations between caption words and images in $\boldsymbol{I_{core}}$.

In our experiments, we use the same 10 sets of images from Corel that are also used in previous work (Duygulu, Barnard et al. 2002; Barnard, Duygulu et al. 2003), so that our results can be compared to the previous results. In the following, the 10 captioned image sets are referred to as the "001", "002", ..., "010" sets. Each of the 10 data sets has around 5,200 images, and each image has about 4 caption words. These images are also called the *core images* from which we try to detect the correlations. For evaluation, each data set is accompanied with a non-overlapping test set $\boldsymbol{I_{test}}$ of around 1,750 images for evaluating the captioning performance. Each test image has the ground truth caption.

Similar to previous work (Duygulu, Barnard et al. 2002; Barnard, Duygulu et al. 2003), each image is represented by a set of image regions. Image regions are extracted using a standard segmentation tool (Shi and Malik 2000), and each region is represented as a 30-dimensional feature vector. The regional features include the mean and standard deviation of RGB values, average responses to various texture filters, its position in the entire image layout, and some shape descriptors (e.g., major orientation and the area ratio of bounding region to the real region). Together, regions extracted from an image form a set-valued attribute "regions" of the object "image". In our experiments, an image has 10 regions on average. Some examples of image regions are shown in Figure 1 (d), (e), and (f).

The exact region segmentation and feature extraction details are *orthogonal* to our approach - any published segmentation methods and feature extraction functions

(Faloutsos 1996) will suffice. All our MAGIC method needs is a black box that will map each color image into a set of zero or more feature vectors.

We want to stress that there is no given information about which region is associated with which term - all we know is that a set of regions co-occurs with a set of terms in an image. That is, no alignment information between individual regions and terms is available.

Therefore, a captioned image becomes an object with two set-valued attributes: “regions” and “terms”. Since the regions and terms of an image are correlated, we propose to use MAGIC to detect this correlation and use it to predict the missing caption terms correlated with the uncaptioned test images.

The first step of MAGIC is to construct the MAGIC graph. Following the instructions for graph construction in Figure 3, the graph for captioned images with attributes “regions” and “terms” will be a 3-layer graph with nodes for images, regions and terms. To form the NN-links, we define the distance function (Assumption 1) between two regions (tokens) as the L_2 norm between their feature vectors. Also, we define that two terms are similar if and only if they are identical, i.e., no term is any other's neighbor. As a result, there is no NN-link between term nodes.

For results shown in this section, the number of nearest neighbors between attribute (domain) tokens is k=3. However, as we will show later in the experimental result section, the captioning accuracy is insensitive to the choice of k. In total, each data set has about 50,000 different region tokens and 160 words, resulting in a graph G_{magic} with about 55,500 nodes and 180,000 edges. The graph based on the core image set $\boldsymbol{I_{core}}$ captures the correlations between regions and terms. We call such graph the “*core*” *graph.*

How do we caption a new image, using the information in a MAGIC graph? Similar to the core images, an uncaptioned image I_{new} is also an object with set-valued attributes: “regions” and “caption”, where attribute “caption” has null value. To find caption words correlated with image I_{new}, we propose to look at regions in the core image set that are similar to the regions of I_{new}, and find the words that are correlated with these core image regions. Therefore, our algorithm has two main steps: finding similar regions in the core image set (augmentation) and identifying caption words (RWR). Next, we define “core graph”, “augmentation” and “gateway nodes”, to facilitate the description of our algorithm.

Definition 4 (Core graph, Augmentation and Gateway Nodes) *For automatic image captioning, we define the* ***core*** *of the graph G_{magic}, G_{core}, be the subgraph that constitutes information of the given captioned images $\boldsymbol{I_{core}}$. The graph G_{magic} for captioning a test image I_{new} is an* ***augmented graph****, which is the core G_{core} augmented with the region-nodes and image-node of I_{new}. The augmentation subgraph is denoted as G_{aug}, and hence the overall G_{magic}=$G_{core} \cup G_{aug}$. The nodes of the core subgraph G_{core} that are adjacent to the augmentation G_{aug} are called the* ***gateway*** *nodes,* ***GW****.*

As an illustration, Figure 2 shows the graph G_{magic} for two core (captioned) images $\boldsymbol{I_{core}}=\{I_1, I_2\}$ and one test (to-be-captioned) image $\boldsymbol{I_{test}}=\{I_3\}$, with the parameter for NN-links k=1. The core subgraph G_{core} contains region nodes $\{r_1, ..., r_7\}$, image nodes $\{I_1, I_2\}$, and all the term nodes $\{t_1, ..., t_8\}$. The augmentation G_{aug} contains region nodes $\{r_8, ..., r_{11}\}$ and the image node $\{I_3\}$ of the test image. The gateway nodes $\boldsymbol{GW} = \{r_5, r_6, r_7\}$ that bridge subgraphs G_{core} and G_{aug} are the nearest neighbors of the test image's regions $\{r_8, ..., r_{11}\}$.

In our experiments, the gateway nodes are always region-nodes in G_{core} that are the nearest neighbors of test image's regions. Different test images have different augmented graphs and gateway nodes. However, since we will caption only one test image at a time, we will use the symbols G_{aug} and $\boldsymbol{GW}$ to represent the augmented graph and gateway nodes of the test image in question.

To sum up, for image captioning, the proposed method first constructs the core graph G_{core}, according to the given set of captioned images $\boldsymbol{I_{core}}$. Then, each test image I_{new} is captioned one by one, in two steps: augmentation and RWR. In the augmentation step, the G_{aug} subgraph of the test image I_{new} is connected to G_{core} via the gateway nodes - the k nearest neighbors of each region of I_{new}. In the RWR step, we do RWR on the whole augmented graph $G_{magic}=G_{core} \cup G_{aug}$, restarting from the test image-node, to identify the correlated words (term-nodes). The g term-nodes with highest RWR scores will be the predicted caption for image I_{new}. Figure 6 gives the details of our algorithm for image captioning.

Input:
1. The core graph G_{core}, an image I_{new} to be captioned.
2. g: The number of caption words we want to predict for I_{new}.

Output: Predicted caption words for I_{new}.

Steps:
1. Augment the image node and region nodes of I_{new} to the core graph G_{core}.
2. Do RWR from the image node of I_{new} on the augmented graph G_{magic} (Algorithm 4).
3. Rank all term nodes by their RWR scores.
4. The g top-ranked terms will be the output - the predicted caption for I_{new}.

Figure 6: The proposed steps for image captioning, using the MAGIC framework.

Captioning Accuracy

We measure captioning performance by the captioning accuracy. The captioning accuracy is defined as the fraction of terms correctly predicted. Following the same evaluation procedure as that in previous work (Duygulu, Barnard et al. 2002; Barnard, Duygulu et al. 2003), for a test image which has g ground-truth caption terms, MAGIC will also predict *g* terms. If *p* of the predicted terms are correct, then the captioning accuracy *acc* on this test image is defined as

$$acc = p/g.$$

The average captioning accuracy acc_{mean} on a set of *T* test images is defined as

$$acc_{mean} = \frac{1}{T}\sum_{i=1}^{T} acc_i \ ,$$

where acc_i is the captioning accuracy on the *i*-th test image.

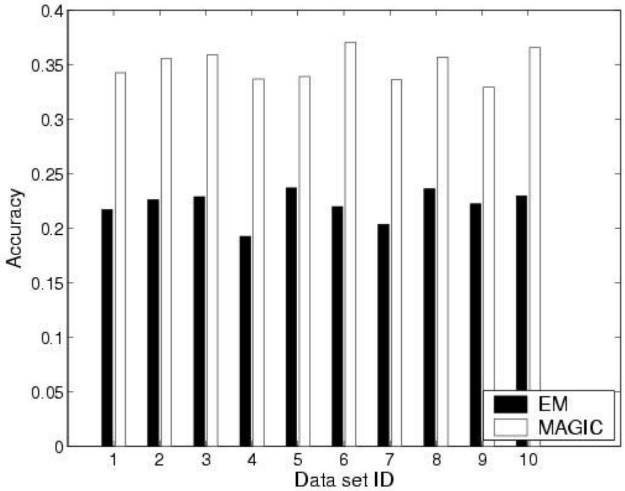


Figure 7: Comparing MAGIC to the EM method. The parameters for MAGIC are *c*=0.66 and *k*=3. The x-axis shows the 10 data sets, and the y-axis is the average captioning accuracy over all test images in a set.

Figure 7 shows the average captioning accuracy on the 10 image sets. We compare our results with the method in (Duygulu, Barnard et al. 2002), which considers the image captioning problem as a statistical translation modeling problem and solves it using expectation-maximization (EM). We refer to their method as the “EM” approach. The x-axis groups the performance numbers of MAGIC (white bars) and EM (black bars) on the 10 data sets. On average, MAGIC achieves captioning accuracy improvement of 12.9 percentage points over the EM approach, which corresponds to a relative improvement of 58%.

The EM method assumes a generative model of caption words given an image region. The model assumes that each region in an image is considered separately when the caption words for an image are “generated”. In other words, the model does not take into account the potential correlations among the “same-image regions” -- regions from a same image. On the other hand, MAGIC incorporates such correlations, by connecting nodes of the “same-image regions” to the same image node in the MAGIC graph. Ignoring the correlations between “same-image regions” could be a reason why the EM method performs not as good as MAGIC.

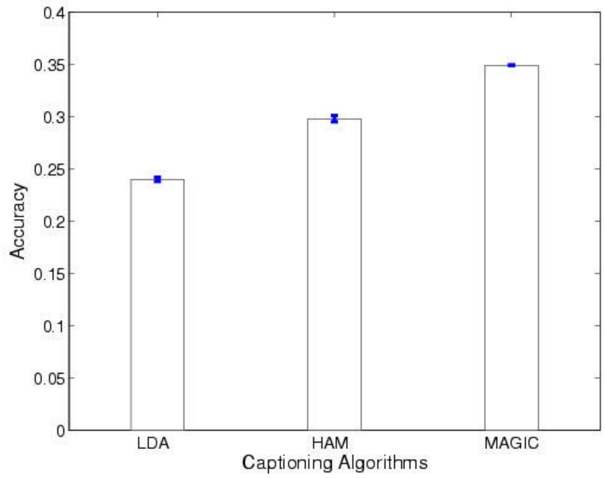


Figure 8: Comparing MAGIC with LDA and HAM. The mean and variance of the average accuracy over the 10 Corel data sets are shown at the y-axis - LDA: (μ, $\sigma2$)=(0.24,0.002); HAM: (μ, σ^2)=(0.298,0.003); MAGIC: (μ, σ^2)=(0.3503, 0.0002). μ: mean average accuracy. σ^2: variance of average accuracy. The range of the error bars at the top of each bar is 2σ.

We also compare the captioning accuracy with even more recent machine vision methods: the Hierarchical Aspect Models method ("HAM") and the Latent Dirichlet Allocation model ("LDA") (Barnard, Duygulu et al. 2003). The methods HAM and LDA are applied to the same 10 Corel data sets, and the average captioning accuracy (averages over the test images) from each set is computed. We summarize the overall performances of a method by taking the mean and variance of the 10 average captioning accuracy values on the 10 data sets. Figure 8 compares MAGIC with LDA and HAM, in terms of the mean and variance of the average captioning accuracy over the 10 data sets. Although both HAM and LDA improve on the EM method, they both lose to our generic MAGIC approach (35%, versus 29% and 25%). It is also interesting that MAGIC gives significantly lower variance, by roughly an order of magnitude: 0.002 versus 0.02 and 0.03. A lower variance indicates that the proposed MAGIC method is more robust to variations among different data sets.

The EM, HAM, and LDA methods all assume a generative model on the relationship among image regions and caption words. For these models, the quality of the data correlations depends on how good the assumed model matches the real data characteristics. Lacking correct insights to the behavior of a data set when designing the model may hurt the performance of these methods.

Unlike EM, HAM, and LDA, which are methods specifically designed for image captioning, MAGIC is a method for general correlation detection. We are pleasantly surprised that a generic method like MAGIC could outperform those domain-specific methods.

Figure 9 shows some examples of the captions given by MAGIC. For the test image I_3 in Figure 1, MAGIC captions it correctly (Figure 9 (a)). In Figure 9(b), MAGIC surprisingly gets the seldom-used word “mane” correctly; however, it mixes up “buildings” with “tree” for the image in Figure 9(c).

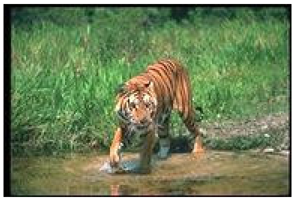
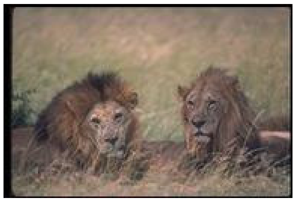
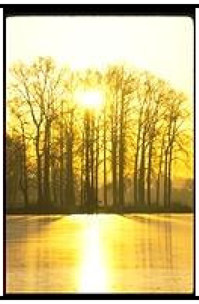
			
	(a)	(b)	(c)
Truth	cat, grass, tiger, water	mane, cat, lion, grass	sun, water, tree, sky
MAGIC	grass, cat, tiger, water	lion, grass, cat, mane	tree, water, buildings, sky

Figure 9: Image captioning examples – terms with highest RWR scores are listed first.

Generalization

MAGIC treats information from all media uniformly as nodes in a graph. Since all nodes are basically the same, we can do RWR and restart from any subset of nodes of any medium, to detect any-to-any medium correlations. The flexibility of our graph-based framework also enables novel applications, such as captioning images in groups (*group captioning*). In this subsection, we show results on (a) detecting the term-to-term correlation in image captioning data sets, and (b) group captioning.

Our image captioning experiments show that MAGIC successfully exploits the image-to-term correlation in the MAGIC graph (G_{magic}) for image captioning. However, the MAGIC graph G_{magic} contains correlations among all media (image, region, and term), not just between images and terms. To show how well MAGIC works on discovering correlations among all media, we design an experiment to extract the term-to-term correlation in the graph G_{magic} and identify correlated captioning terms.

We use the same 3-layer MAGIC core graph G_{core} constructed in the previous subsection for automatic image captioning (Figure 2). Given a query term *t*, we use RWR to find other terms correlated with it. Specifically, we perform RWR, restarting from the query term(-node). The terms whose corresponding term-nodes receive high RWR scores are considered correlated with the query term.

Table 2 shows the top 5 terms with the highest RWR scores for some query terms. In the table, each row shows a query term at the first column, followed by the top 5 correlated terms selected by MAGIC (sorted by their RWR scores). The selected terms have meanings that are semantically related with the query term. For example, the term “branch”, when used in image captions, is strongly related to forest- or bird- related concepts. MAGIC shows exactly this, correlating “branch” with terms such as “birds”, “owl”, and “nest”.

Query term	1	2	3	4	5
branch	birds	night	owl	nest	hawk
bridge	water	arch	sky	stone	boats
cactus	saguaro	desert	sky	grass	sunset
car	tracks	street	buildings	turn	prototype
f-16	plane	jet	sky	runway	water
market	people	street	food	closeup	buildings
mushrooms	fungus	ground	tree	plants	coral
pillars	stone	temple	people	sculpture	ruins
reefs	fish	water	ocean	coral	sea
textile	pattern	background	texture	designs	close-up

Table 2: Correlated terms of some query terms.

Another subtle observation is that our method does not seem to be biased by frequently appeared words. In our collection, the terms “water” and “sky” appear more frequently in image captions, i.e., they are like terms “the” and “a” in normal English text. Yet, these frequent terms do *not* show up too often in Table 2, as a correlated term of a query term. It is surprising, given that we do nothing special when using MAGIC: no tf/idf weighting, no normalization, and no other domain-specific analysis. We just treat these frequent terms as nodes in our MAGIC graph, like any other nodes.

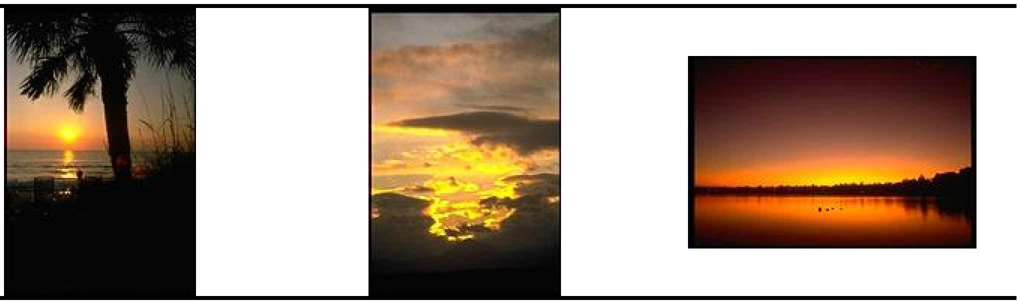
Another advantage of the proposed MAGIC method is that it can be easily extended to caption a group of images, considering the whole group at once. This flexibility is due to the graph-based framework of MAGIC, which allows augmentation of multiple nodes and doing RWR from any subset of nodes. To the best of our knowledge, MAGIC is the first method that is capable of doing group captioning.

Application 2 (Group Captioning) *Given a set* $\boldsymbol{I_{core}}$ *of captioned images and a (query) group of uncaptioned images* $\{I'_1, ..., I'_t\}$*, find the best g (say, g=5) caption words to assign to the group.*

Possible applications for group captioning include video segment captioning, where a video segment is captioned according to the group of keyframes associated with the segment. Since keyframes in a video segment are usually related, captioning them as a whole can take into account the inter-keyframe correlations, which are missed if each keyframe is captioned separately. Accurate captions for video segments may improve performances on tasks such as video retrieval and classification.

The steps to caption a group of images are similar to those for the single-image captioning outlined in Figure 6. A core MAGIC graph is still used to capture the mixed media information of a given collection of captioned images. The different steps for doing group captioning are: First, instead of augmenting the single query image to the core and restarting from it, we augment all *t* images in the query group $\{I'_1, ..., I'_t\}$ to the core. Then, the RWR step is performed by randomly restarts from one of the images in the group (i.e., each of the *t* query image has probability *1/t* to be the restart node).

Figure 10 shows the result of using MAGIC for captioning a group of three images. MAGIC finds reasonable terms for the entire group of images: “sky”, “water”, “tree”, and “sun”. Captioning multiple images as a group takes into consideration the correlations between different images in the group, and in this example, this helps reduce the scores of irrelevant terms such as “people”. In contrast, when we caption these images individually, MAGIC selects “people” as caption words for images in Figure 10(a) and (b), which do not contain people-related objects.



	(a)	(b)	(c)
Truth	sun, water, tree, sky	sun, clouds, sky, horizon	sun, water
MAGIC	sky, water, tree, sun		

Figure 10: Group captioning – caption terms with highest RWR scores are listed first.

SYSTEM ISSUES

MAGIC provides an intuitive framework for detecting cross-modal correlations. The RWR computation in MAGIC is fast and it scales linearly with the graph size. For example, a straightforward implementation of RWR can caption an image in less than 5 seconds.

In this section, we discuss system issues such as parameter configuration and fast computation. In particular, we present results showing

- MAGIC is insensitive to parameter settings, and
- MAGIC is modular that we can easily employ the best module to date to speedup MAGIC.

Optimization of Parameters

There are several design decisions to be made when employing MAGIC for correlation detection: *what should be the values for the two parameters: the number of neighbors k of a domain token, and the restart probability c of RWR?* And, *should we assign weights to edges, according to the types of their end points?* In this section, we empirically show that the performance of MAGIC is insensitive to these settings, and provide suggestions on determining reasonable default values.

We use automatic image captioning as the application to measure the effect of these parameters. The experiments in this section are performed on the same 10 captioned image sets ("001", ..., "010") described in previous sections, and we measure how the values of the parameters, such as *k*, *c*, and the weights of the links of the MAGIC graphs, effect the captioning accuracy.

Number of Neighbors *k*. The parameter *k* specifies the number of nearest domain tokens to which a domain token connects via the NN-links. With these NN-links, objects having little difference in their attribute values will be closer to each other in the graph, and therefore, are considered more correlated by MAGIC. For *k*=0, all domain tokens are considered distinct; for larger *k*, our application is more tolerant to the difference in attribute values.

We examine the effect of various *k* values on image captioning accuracy. Figure 11 shows the captioning accuracy on the data set "006", with the restart probability *c*=0.66. The captioning accuracy increases as *k* increases from *k*=1, and reaches a plateau between *k*=3 and 10. The plateau indicates that MAGIC is insensitive to the value of *k*. Results on other data sets are similar, showing a plateau between *k*=3 and 10.

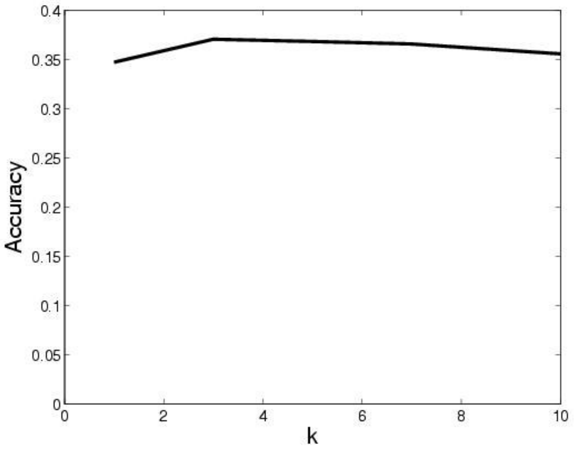


Figure 11: The plateau in the plot shows that the captioning accuracy is insensitive to value of the number of nearest neighbors *k*. Y-axis: Average accuracy over all test images of data set "006". The restart probability is *c*=0.66.

In hindsight, with only *k*=1, the collection of regions (domain tokens) is barely connected, missing important connections and thus leading to poor performance on detecting correlations. At the other extreme, with a high value of *k*, everybody is directly connected to everybody else and there is no clear distinction between really close neighbors or just neighbors. For a medium value of *k*, the NN-links apparently capture the correlations between the close neighbors and avoid noise from remote neighbors. Small deviations from that value make little difference, which is probably because that the extra neighbors we add (when *k* increases), or those we retained (when *k* decreases), are at least as good as the previous ones.

Restart Probability *c*. The restart probability *c* specifies the probability to jump back to the restarting node(s) of the random walk. Higher value of *c* implies giving higher RWR scores to nodes closer in the neighborhood of the restart node(s). Figure 12 shows the image captioning accuracy of MAGIC with different values of *c*. The data set is "006", with the parameter *k*=3. The accuracy reaches a plateau between *c*=0.5 and 0.9, showing that the proposed MAGIC method is insensitive to the value of *c*. Results on other data sets are similar, showing a plateau between *c*=0.5 and 0.9.

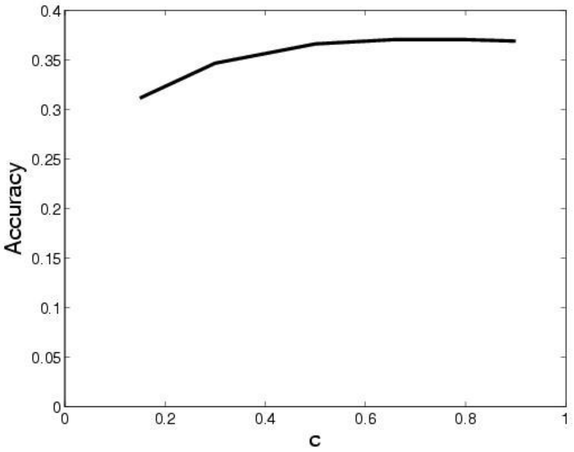


Figure 12: The plateau in the plot shows that the captioning accuracy is insensitive to value of the restart probability *c*. Y-axis: Average accuracy over all images of data set "006". The number of nearest neighbors per domain token (region) is *k*=3.

For web graphs, the recommended value for *c* is typically *c*=0.15 (Haveliwala, Kamvar et al. 2003). Surprisingly, our experiments show that this choice does not give good performance. Instead, good quality is achieved for *c*=0.6~0.9. Why is this discrepancy?

We conjecture that what determines a good value for the restart probability is the diameter of the graph. Ideally, we want our random walker to have a non-trivial chance to reach the outskirts of the whole graph. If the diameter of the graph is *d*, the probability that the random walker (with restarts) will reach a point on the periphery is proportional to $(1\text{-}c)^d$, i.e., the probability of not restarting for *d* consecutive moves.

For the web graph, the diameter is estimated to be *d*=19 (Albert, Jeong et al. 1999). This implies that the probability $p_{periphery}$ for the random walker to reach a node in the periphery of the web graph is roughly

$$p_{periphery} = (1\text{-}c)^{19} = (1\text{-}0.15)^{19} = 0.045 .$$

In our image captioning experiments, we use graphs that have three layers of nodes (Figure 2). The diameter of such graphs is roughly *d*=3. If we demand the same $p_{periphery}$=0.045, then the *c* value for our 3-layer graph would be

$$(1\text{-}0.15)^{19} = (1\text{-}c)^3,$$

$$\Rightarrow c = 0.65,$$

which is much closer to our empirical observations. Of course, the problem requires more careful analysis - but we are the first to show that c=0.15 is not always optimal for random walk with restarts.

Link weights. MAGIC uses a graph to encode the relationship between mixed media objects and their attributes of different media. The OAV-links in the graph connect objects to their domain tokens (Figure 2). To give more attention to an attribute domain D, we can increase the weights of OAV-links that connect to the tokens of domain D. *Should we treat all media equally, or should we weight OAV-links according to their associated domains? How should we weight the OAV-links? Could we achieve better performance on weighted graphs?*

	w_{region}		
w_{term}	0.1	1	10
0.1	0.370332	0.371963	0.370812
1	0.369900	0.370524	0.371963
10	0.368969	0.369181	0.369948

Table 3: Captioning accuracy is insensitive to various weight settings on OAV-links to the two media: region (weight w_{region}) and term (w_{term}).

We investigate how the change on link weights influences the image captioning accuracy. Table 3 shows the captioning accuracy on data set "006" when different weights are assigned on the OAV-links to regions (weight w_{region}) and those to terms (w_{term}). Specifically, the elements $A_{i,j}$ of the adjacency matrix $\mathbf{A}$ will now take values w_{region} or w_{term}, besides values 0 and 1, depending on the type of link $A_{i,j}$ corresponds to. For all cases, the number of nearest neighbors is k=3 and the restart probability is c=0.66. The case where (w_{region}, w_{term})=(1,1) is that of the unweighted graph, and is the result we reported in Figure 7. As link weights vary from 0.1, 1 to 10, the captioning accuracy is basically unaffected. The results on other data sets are similar – captioning accuracy is at the same level on a weighted graph as on the unweighted graph.

This experiment shows that an unweighted graph is appropriate for our image captioning application. We speculate that an appropriate weighting for an application depends on properties such as the number of attribute domains (i.e., the number of layers in the graph), the average size of a set-valued attribute of an object (such as, average number of regions per image), and so on. We plan to investigate more on this issue in our future work.

Speedup Graph Construction by Approximation

The proposed MAGIC method encodes a mixed media data set as a graph, and employs the RWR algorithm to find cross-modal correlations. The construction of the G_{magic} graph is intuitive and straightforward, and the RWR computation is light and linear to the database size. One step that is relatively expensive is the construction of NN-links in a MAGIC graph.

When constructing the NN-links of a MAGIC graph, we need to compute the nearest neighbors for every domain token. For example, in our image captioning experiments, to construct the NN-links among region-nodes, *k*-NN searches are performed about 50,000 times (one for each region token) in the 30-dimensional region-feature space.

In MAGIC, the NN-links are proposed to capture the similarity relation among domain tokens. The goal is to associate similar tokens to each other, and therefore, it could be suffice to have the NN-links connect to neighbors that are close enough, even if they are not exactly the closest ones. The approximate nearest neighbor search is usually faster, by trading accuracy for speed. The interesting questions are: *How much speedup could we gain by allowing approximate NN-links? How much is the performance reduction by approximation?*

For efficient nearest neighbor search, one common way is to use a spatial index, such as R^+-tree (Sellis, Roussopoulos et al. 1987), to find exact nearest neighbors in logarithmic time. Fortunately, MAGIC is modular and we can pick the best module to perform each step. In our experiments, we use the approximate nearest neighbor searching (ANN) (Arya, Mount et al. 1998), which supports both exact and approximate nearest neighbor search. ANN estimates the distance to a nearest neighbor up to $(1+\varepsilon)$ times of the actual distance: $\varepsilon=0$ means exact search with no approximation; bigger ε values give rougher estimation.

	Approximate Nearest Neighbor Search			Sequential
	ε=0	ε=0.2	ε=0.8	search (SS)
Elapse time (msec.)	3.8	2.4	0.9	46
Speedup to SS	12.1	19.2	51.1	1
Error (in top *k*=10)	-	0.0015%	1.67%	-
Error (in top *k*=3)	-	-	0.46%	-

Table 4: The efficiency/accuracy trade off of constructing NN-links using an approximate method (ANN). ε=0 indicates the exact nearest neighbor computation. Elapse time: average wall clock time for one nearest neighbor search. Speedup: the ratio of the time of sequential search (SS) and that of an ANN search. The error is measured as the percentage of mistakes made by approximation on the *k* nearest neighbors. The symbol "-" means zero error.

Table 4 lists the average wall clock time to compute the top 10 neighbors of a region in the 10 Corel image sets of our image captioning experiments. The table shows the efficiency/accuracy trade off on constructing the NN-links among image regions. In an approximate nearest neighbor search, the distance to a neighboring region is approximated to within $(1+\varepsilon)$ times of the actual distance. The speedup of using the approximate method is compared to the sequential search method (SS). In our experiments, the sequential search method is implemented in C++ and compiled with the code optimization, using the command "g++ -O3".

Compared to the sequential search, the speedup of using the ANN method increases from 12.1 to 51.1, from the exact search (ε=0) to a rough approximation of (ε=0.8). For the top k=3 nearest neighbors (the setting used in most of our experiments), the error percentage is at most 0.46% for the roughest approximation, which is equivalent to making one error in every 217 NN-links.

We conduct experiments to investigate the effect on captioning accuracy from using the approximate NN-links. In general, the small differences on NN-links due to approximation do not change the characteristic of the MAGIC graph significantly, and has limited affect on the performance of image captioning (Figure 13). At the approximation level ε=0.2, we achieve a speedup of 19.1 times and surprisingly that no error is made on the NN-links in the MAGIC graph, and therefore the captioning accuracy is the same as exact computation. At the approximation level ε=0.8, which gives an even better speedup of 51.1 times, the average captioning accuracy decreases by just 1.59 percentage point (averaged over the 10 Corel image sets). Therefore, by using an approximate method, we can significantly reduce the time to construct the MAGIC graph (up to 51.1 times speedup), with almost no decrease on the captioning accuracy.

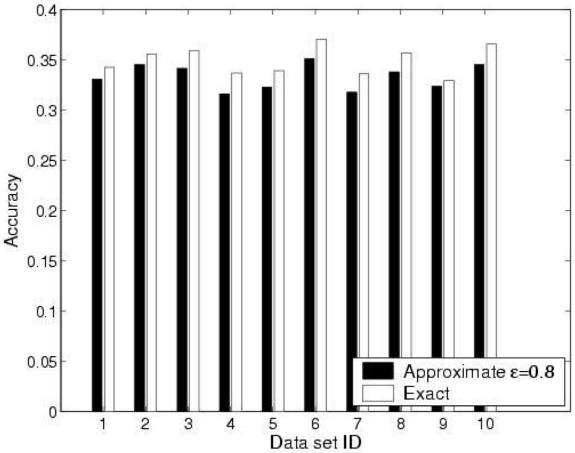


Figure 13: Speeding up NN-links construction by ANN (with ε=0.8) reduces captioning accuracy by just 1.59% on the average. X-axis: 10 data sets. Y-axis: average captioning accuracy over test images in a set. In this experiment, the parameters are c=0.66 and k=3.

CONCLUSIONS

Mixed media objects such as captioned images or video clips contain attributes of different modalities (image, text, or audio). Correlations across different modalities provide information about the multimedia content, and are useful in applications ranging from summarization to semantic captioning. In this paper, we develop MAGIC, a graph-based method for detecting cross-modal correlations in mixed media data set.

There are two challenges in detecting cross-modal correlations, namely, representation of attributes of various modalities and the detection of correlations among any subset of modalities. MAGIC turns the multimedia problem into a graph problem, and provides an intuitive solution that easily incorporates various modalities. The graph framework of MAGIC creates opportunity for applying graph algorithms to multimedia problems. In particular, MAGIC finds cross-modal correlations using the technique of random walk with restarts (RWR), which accommodates set-valued attributes and noise in data with no extra effort.

We apply MAGIC for automatic image captioning. By finding robust correlations between text and image, MAGIC achieves a relative improvement by 58% in captioning accuracy as compared to recent machine learning techniques (Figure 8). Moreover, the MAGIC framework enables novel data mining applications, such as *group captioning* where multiple images are captioned simultaneously, taking into account the possible correlations between the multiple images in the group (Figure 10).

Technically, MAGIC has the following desirable characteristics:

- It is domain independent: The $Sim_i(*,*)$ similarity functions (Assumption 1) completely isolate our MAGIC method from the specifics of an application domain, and make MAGIC applicable to detect correlations in all kinds of mixed media data sets.
- It requires no fine-tuning on parameters or link weights: The performance is not sensitive to the two parameters - the number of neighbors *k* and the restart probability *c*, and it requires no special weighting scheme like tf/idf for link weights.
- Its computation is fast and scales up well with the database/graph size.
- It is modular and can easily incorporate recent advances in related areas (e.g., fast nearest neighbor search) to improve performance.

We are pleasantly surprised that such a domain-independent method, with no parameters to tune, outperforms some of the most recent and carefully tuned methods for automatic image captioning. Most of all, the graph-based framework proposed by MAGIC creates opportunity for applying graph algorithms to multimedia problems. Future work could further exploit the promising connection between multimedia databases and graph algorithms for other data mining tasks, including multi-modal event summarization (Pan, Yang et al. 2004c) or outlier detection, that require the discovery of correlations as its first step.

REFERENCES

Albert, A., H. Jeong, et al. (1999). Diameter of the World Wide Web. Nature. **401:** 130-131.

Arya, S., D. M. Mount, et al. (1998). An optimal algorithm for approximate nearest neighbor searching. Journal of the ACM. **45:** 891-923.

Barnard, K., P. Duygulu, et al. (2003). Matching words and pictures. Journal of Machine Learning Research. **3:** 1107-1135.

Benitez, A. B. and S.-F. Chang (2002). Multimedia Knowledge Integration, Summarization and Evaluation. Proceedings of the 2002 International Workshop on Multimedia Data Mining in conjunction with the International Conference on Knowledge Discovery and Data Mining (MDM/KDD-2002). Edmonton, Alberta, Canada**:** 39-50.

Blei, D. M. and M. I. Jordan (2003). Modeling Annotated Data. Proceedings of the 26th annual International ACM SIGIR Conference on Research and Development in Information Retrieval**:** 127-134.

Brin, S. and L. Page (1998). The anatomy of a large-scale hypertextual web search engine. Proceedings of the Seventh International Conference on World Wide Web. Brisbane, Australia**:** 107-117.

Chang, S.-F., R. Manmatha, et al. (2005). Combining Text and Audio-Visual Features in Video Indexing. Proceedings of the 2005 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2005). Philadelphia, PA, USA. **5:** 1005-1008.

Doyle, P. G. and J. L. Snell (1984). Random Walks and Electric Networks, Mathematical Association of America.

Duygulu, P., K. Barnard, et al. (2002). Object recognition as machine translation: learning a lexicon for a fixed image vocabulary. Proceedings of the Seventh European Conference on Computer Vision (ECCV 2002). **4:** 97-112.

Duygulu, P., J.-Y. Pan, et al. (2004). Towards Auto-Documentary: Tracking the Evolution of News Stories. Proceedings of the annual International ACM Conference on Multimedia. New York, NY, USA**:** 820-827.

Faloutsos, C. (1996). Searching Multimedia Databases by Content, Kluwer Academic Publishers.

Feng, S. L., R. Manmatha, et al. (2004). Multiple Bernoulli Relevance Models for Image and Video Annotation. Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2004). **2:** 1002-1009.

Haveliwala, T., S. Kamvar, et al. (2003). An Analytical Comparison of Approaches to Personalizing PageRank. (Tech. Rep. No. 2003-35). InfoLab, Stanford University, CA, USA.

Haveliwala, T. H. (2002). Topic-Sensitive PageRank. Proceedings of the 11th International Conference on World Wide Web. Honolulu, Hawaii, USA**:** 517-526.

Hsu, W., L. Kennedy, et al. (2004). News Video Story Segmentation using Fusion of Multi-Level Multi-modal Features in TRECVID 2003. Proceedings of the 2004 IEEE

International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2004). Montreal, Quebec, Canada. **3:** 645-648.

Jeon, J., V. Lavrenko, et al. (2003). Automatic Image Annotation and Retrieval using Cross-Media Relevance Models. Proceedings of the 26th annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Toronto, Canada**:** 119-126.

Jin, R., J. Y. Chai, et al. (2004). Effective automatic image annotation via a coherent language model and active learning. Proceedings of the 12th annual ACM international conference on Multimedia. New York, NY, USA**:** 892 - 899.

Kamvar, S. D., T. H. Haveliwala, et al. (2003a). Adaptive Methods for the Computation of PageRank. Proceedings of the International Conference on the Numerical Solution of Markov Chains (NSMC)**:** 31-44.

Kamvar, S. D., T. H. Haveliwala, et al. (2003). Extrapolation Methods for Accelerating PageRank Computation. Proceedings of the 12th International Conference on World Wide Web. Budapest, Hungary**:** 261-270.

Kleinberg, J. (1998). Authoritative sources in a hyperlinked environment. Proceedings of the 9th annual ACM-SIAM Symposium on Discrete Algorithms. San Francisco, CA, USA**:** 668-677.

Li, D., N. Dimitrova, et al. (2003). Multimedia content processing through cross-modal association. Proceedings of the eleventh ACM international conference on Multimedia. Berkeley, CA, USA**:** 604-611.

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

Lin, W.-H. and A. Hauptmann (2002). News Video Classification Using SVM-based Multimodal Classifiers and Combination Strategies. Proceedings of the 10th annual ACM international conference on Multimedia. Juan-les-Pins, France**:** 323-326.

Liu, W., S. Dumais, et al. (2001). Semi-Automatic Image Annotation. Proceedings of the 8th IFIP TC.13 Conference on Human-Computer Interaction (INTERACT 2001).

Lovász, L. (1996). Random Walks on Graphs: A Survey. Combinatorics, Paul Erdös is Eighty. **2:** 353-398.

Maron, O. and A. L. Ratan (1998). Multiple-Instance Learning for Natural Scene Classification. Proceedings of the Fifteenth International Conference on Machine Learning**:** 341-349.

Mori, Y., H. Takahashi, et al. (1999). Image-to-word transformation based on dividing and vector quantizing images with words. Proceedings of the First International Workshop on Multimedia Intelligent Storage and Retrieval Management. Orlando, Florida, USA.

Naphade, M. R., I. Kozintsev, et al. (2001). Probabilistic Semantic Video Indexing. Advances in Neural Information Processing Systems (NIPS) Denver, CO, USA. **13**.

Palmer, C. R. and C. Faloutsos (2003). Electricity Based External Similarity of Categorical Attributes. Proceedings of the 7th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD 2003). Seoul, South Korea**:** 486-500.

Pan, J.-Y. and C. Faloutsos (2002). VideoCube: a novel tool for video mining and classification. Proceedings of the Fifth International Conference on Asian Digital Libraries (ICADL 2002). Singapore**:** 194-205.

Pan, J.-Y., H.-J. Yang, et al. (2004). Automatic Image Captioning. Proceedings of the 2004 IEEE International Conference on Multimedia and Expo (ICME 2004). Taipei, Taiwan. **3:** 1987-1990.

Pan, J.-Y., H.-J. Yang, et al. (2004a). Automatic Multimedia Cross-modal Correlation Discovery. Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Seattle, WA, USA**:** 653-658.

Pan, J.-Y., H.-J. Yang, et al. (2004b). GCap: Graph-based Automatic Image Captioning. Proceedings of the 2004 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW 2004). **9:** 146.

Pan, J.-Y., H. Yang, et al. (2004c). MMSS: Multi-modal Story-oriented Video Summarization. Proceedings of the Fourth IEEE International Conference on Data Mining (ICDM 2004). Brighton, UK**:** 491-494.

Sebe, N., M. S. Lew, et al. (2003). The State of the Art in Image and Video Retrieval. Proceedings of the International Conference on Image and Video Retrieval (CIVR'03). Urbana, IL, USA**:** 1-8.

Sellis, T. K., N. Roussopoulos, et al. (1987). The R^+-Tree: A Dynamic Index for Multi-Dimensional Objects. Proceedings of the 12th International Conference on Very Large Data Bases**:** 507-518.

Shi, J. and J. Malik (2000). Normalized Cuts and Image Segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence. **22:** 888-905.

Srihari, R. K., A. Rao, et al. (2000). A Model for Multimodal Information Retrieval. Proceedings of the 2000 IEEE International Conference on Multimedia and Expo (ICME 2000). **2:** 701-704.

Virga, P. and P. Duygulu (2005). Systematic Evaluation of Machine Translation Methods for Image and Video Annotation. Proceedings of the Fourth International Conference on Image and Video Retrieval (CIVR 2005). Singapore**:** 174-183.

Vries, A. P. d., T. Westerveld, et al. (2004). Combining multiple representations on the TRECVID search task. Proceedings of 2004 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2004). Montreal, Quebec, Canada. **3:** 1052-1055.

Wang, X.-J., W.-Y. Ma, et al. (2004). Multi-model similarity propagation and its application for web image retrieval. Proceedings of the 12th annual ACM international conference on Multimedia. New York, NY, USA**:** 944-951.

Wu, Y., E. Y. Chang, et al. (2004). Optimal multimodal fusion for multimedia data analysis. Proceedings of the 12th annual ACM international conference on Multimedia. New York, NY, USA**:** 572-579.

Xie, L., L. Kennedy, et al. (2005). Layered dynamic mixture model for pattern discovery in asynchronous multi-modal streams. Proceedings of the 2005 IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP 2005). Philadelphia, PA, USA. **2:** 1053-1056.

Zhang, D.-Q., C.-Y. Lin, et al. (2004). Semantic Video Clustering Across Sources Using Bipartite Spectral Clustering. Proceeding of 2004 IEEE Conference on Multimedia and Expo (ICME 2004). Taipei, Taiwan. **1:** 117-120.

Zhang, Z., R. Zhang, et al. (2004). Exploiting the cognitive synergy between different media modalities in multimodal information retrieval. Proceedings of the 2004 IEEE International Conference on Multimedia and Expo (ICME 2004). Taipei, Taiwan. **3:** 2227-2230.

Zhao, R. and W. I. Grosky (2001). Bridging the Semantic Gap in Image Retrieval. In T. K. Shih (Ed.), Distributed Multimedia Databases: Techniques and Applications**:** 14-36.