Exploiting Multi-Modal Interactions: A Unified Framework

Ming Li and **Xiao-Bing Xue** and **Zhi-Hua Zhou***
National Key Laboratory for Novel Software Technology
Nanjing University, Nanjing 210093, China
{lim, xuexb, zhouzh}@lamda.nju.edu.cn

Abstract

Given an imagebase with tagged images, four types of tasks can be executed, i.e., content-based image retrieval, image annotation, text-based image retrieval, and query expansion. For any of these tasks the similarity on the concerned type of objects is essential. In this paper, we propose a framework to tackle these four tasks from a unified view. The essence of the framework is to estimate similarities by exploiting the interactions between objects of different modality. Experiments show that the proposed method can improve similarity estimation, and based on the improved similarity estimation, some simple methods can achieve better performances than some state-of-the-art techniques.

1 Introduction

With the explosive accumulation of digital images, many imagebases have been developed. How to organize and search an imagebase effectively and efficiently poses a big challenge for researchers. Previous researches on *content-based image retrieval* (CBIR) suffer from the big gap between the low-level visual feature and the high-level semantics. Recently, much attention has been paid to exploiting some tagged images to bridge such semantic gap. Actually, given a group of tagged images, several types of tasks can be executed on an imagebase, which can be summarized as follows.

- **Image-In Image-Out**: This corresponds to the *content-based image retrieval* task, where the user poses a query image and then the system returns a set of relevant images.
- **Image-In Text-Out**: This corresponds to the *image annotation* task, where the user poses a query image and then the system returns a set of annotation words.
- **Text-In Image-Out**: This corresponds to the *text-based image retrieval* task, where the user poses a text query and then the system returns a set of relevant images.
- **Text-In Text-Out**: This corresponds to the *query expansion* task, where the user poses a text query and then the system returns some words as hints to help the user to refine the original query.

Many successful approaches have been proposed for each type of the above tasks respectively. For example, Translation Model (TM) [Duygulu *et al.*, 2002], Latent Dirichlet Allocation (MoM-LDA) [Barnard *et al.*, 2003] and Hidden Markov Model (2D MHMM) [Li and Wang, 2003] have been used for image annotation. Cross Media Relevance Model (CMRM) [Jeon *et al.*, 2003] and Structure-Composition Model [Datta *et al.*, 2006] have demonstrated their effectiveness for text-based image retrieval. CBIR [Smeulders *et al.*, 2000] has been studied thoroughly in pure imagebases which contain only untagged images, while query expansion [Baeza-Yates and Ribeiro-Neto, 1999] is a hot topic in pure textbases that do not involve images.

However, few studies have considered all these tasks from a unified view. In fact, although these four tasks focus on different goals, they all involve common basic elements, i.e., the blob (the visual feature used to represent an image) , the word (the unit of an image's annotation) and the image itself. Thus, a good estimation of the similarities on these elements is essential for all these tasks.

The conventional similarity between two blobs (words) is based on the co-occurrence in images. In other words, two blobs (words) that tend to appear in the same images are similar. However, this similarity could not account for the situation that two blobs (words) appear in similar but not the same images. Thus, a more reasonable estimation of similarity between blobs (words) needs to consider the similarity between images. Similarly, the similarity between images also relies on the similarity between blobs (words). It is more reasonable to consider two images as similar when they contain similar blobs (words) instead of exactly the same ones. Generally, in imagebase the similarities between one type of objects (e.g., blobs) are inevitably influenced by the similarities between other type of objects (e.g., images). It is evident that an improved similarity on one type of entity may help to improve the similarity on another type of entity. Thus, by working in a unified framework, it may be possible to improve the similarities of all types of objects by exploiting the interactions among them. Experiments show that, amazingly, simple methods with the improved similarity can even beat the

*This research was supported by NSFC (60635030, 60721002), 863 Program (2007AA01Z169), JiangsuSF (BK2008018) and Jiangsu 333 Program. Currently the second author is pursuing Ph.d. degree at the University of Massachusetts at Amherst.

state-of-the-art techniques on the four aforementioned tasks.

Above all, a different strategy is adopted in this paper: instead of designing sophisticated models for each type of the four tasks respectively, we focus on the common basis of these tasks, i.e., the similarity of the concerned objects. The improved estimation of similarity has been obtained through exploiting the interactions among different objects under a unified framework.

The rest of paper is organized as follows. Section 2 briefly reviews some related work. Section 3 proposes the unified framework for exploiting interactions among objects. Section 4 reports experimental results. Finally, Section 5 concludes.

2 Related Work

Many studies have been conducted for image annotation and text-based image retrieval. In addition to the work mentioned in Section 1, Blei and Jordan [2003] proposed the correspondence LDA (Corr-LDA) to model the conditional relationship between the latent variables and Xing *et al.* [2005] proposed the dual-wing harmonium model for image annotation, which bears the properties of efficient inference and robust topic mixing. Kang *et al.* [2006] proposed the Correlated Label Propagation (CLP) to explicitly model the correlations between class labels.

Since CMRM and CLP will be used as baselines in our experiments, here we introduce more details. CMRM is a modification of the relevance based language model. The joint probability of words and blobs are modelled through accumulating the evidence from the training set. After marginalizing this joint probability, the image can be annotated easily. Unlike previous methods which propagate the class labels independently, CLP simultaneously co-propagates multiple labels from the training set to the test set and uses an efficient algorithm to avoid combinatorial explosion.

Kandola *et al.* [2003] proposed the semantic similarity, which models the interaction between document and word, with application to text classification. Lafferty and Zhai [2001] used the Markov chain to model the same interaction for text retrieval.

Hardoon *et al.* [2003] proposed to use Kernel Canonical Correlation Analysis (KCCA) to learn a semantic representation for images with associated text. This semantic representation can be considered as a new feature space, where the correlation between the visual features of images and the associated text of images is maximized. The cross-media retrieval can be conducted through first mapping both the query and images into this semantic space and then calculating the inner-product between the query and images. The major difference between KCCA and our method lies in the fact that the maximization of the correlation between two feature sets does not necessarily lead to the minimization of the difference between two similarity matrices of images calculated from different feature sets. In our method, we implicitly achieve the latter target through modelling the interactions between image and text.

Pan *et al.* [2004] proposed a graph-based method (GCap). After constructing the graph for each type of objects, a random walk with restarts is executed to solve the task in imagebase. Guo *et al.* [2007] explicitly modelled the relations between blob and word by structural learning techniques. Our method focuses on the refined similarities of each type of objects instead of the cross type relations.

3 A Unified Framework

Here, we consider an imagebase consisting of a set of tagged images T, a set of untagged images U, a set of image blobs B and a set of annotation words W.

Formally, $B = \{b_1, b_2, \cdots, b_c\}$ where b_j is an image blob; $W = \{w_1, w_2, \cdots, w_d\}$ where w_j is an annotation word. $T = \{t_1, t_2, \cdots, t_m\}$ where $t_i = (\vec{x}_i, \vec{y}_i)$ is a tagged image, $\vec{x}_i = [\#(b_1, t_i), \#(b_2, t_i), \cdots, \#(b_c, t_i)]^\top$, $\vec{y}_i = [\#(w_1, t_i), \#(w_2, t_i), \cdots, \#(w_d, t_i)]^\top$. Here $\#(b_j, t_i)$ indicates the frequency that the blob b_j appears in t_i, and $\#(b_j, t_i) = 0$ if b_j does not appear in t_i. $\vec{x}_i$ is the visual representation vector of t_i[1]. Similarly, $\vec{y}_i$ is the annotation word vector of t_i; $\#(w_j, t_i)$ indicates the frequency the word w_j appears in the annotation of t_i, and $\#(w_j, t_i) = 0$ if w_j does not appear in t_i. $U = \{u_1, u_2, \cdots, u_l\}$ where $u_i = (\vec{z}_i, \phi)$ is an untagged image, i.e., the annotation words of the image are not known, $\vec{z}_i = [\#(b_1, u_i), \#(b_2, u_i), \cdots, \#(b_c, u_i)]^\top$. It is also possible to describe each blob b_j and each word w_j by considering its appearances in the images in T as $\vec{b}_j = [\#(b_j, t_1), \#(b_j, t_2), \cdots, \#(b_j, t_m)]^\top$ where $\#(b_j, t_i)$ indicates the frequency b_j appears in the image t_i, and $\vec{w}_j = [\#(w_j, t_1), \#(w_j, t_2), \cdots, \#(w_j, t_m)]^\top$ where $\#(w_j, t_i)$ indicates the frequency w_j appears in the annotation of the image t_i[2]. Furthermore, $c \times m$ blob-image relation matrix is denoted as $\mathbf{M} = [\vec{x}_1, \vec{x}_2, \cdots, \vec{x}_m] = [\vec{b}_1, \vec{b}_2, \cdots, \vec{b}_c]^\top$ and the $d \times m$ word-image relation matrix is denoted as $\mathbf{N} = [\vec{y}_1, \vec{y}_2, \cdots, \vec{y}_m] = [\vec{w}_1, \vec{w}_2, \cdots, \vec{w}_d]^\top$[3].

3.1 Interactions of Different Types of Objects

Assume $\mathbf{S_B}$ denotes the similarity matrix of blobs where $\mathbf{S_B}(i, j) = sim(b_i, b_j)$, $\mathbf{S_W}$ denotes the similarity matrix of words where $\mathbf{S_W}(i, j) = sim(w_i, w_j)$, and $\mathbf{S_T}$ denotes the similarity matrix of images where $\mathbf{S_T}(i, j) = sim(t_i, t_j)$. The calculation of the similarities of each type of objects is summarized in Table 1. Note that the similarity of images can be calculated based on either blobs or words they contain.

For example, the first row of Table 1 shows the initial similarity of two blobs[4], which assumes that blobs appearing in the *same* images tend to be similar. By further considering the similarity of images, a refined blob similarity is shown in the second row, which assumes that blobs appearing in *similar* images tend to be similar.

Table 1 clearly shows that the estimations of $\mathbf{S_B}$, $\mathbf{S_W}$ and $\mathbf{S_T}$ are influenced by each other. In order to differentiate

[1] The blob-based representation is widely used for image annotation and retrieval [Duygulu *et al.*, 2002; Jeon *et al.*, 2003].

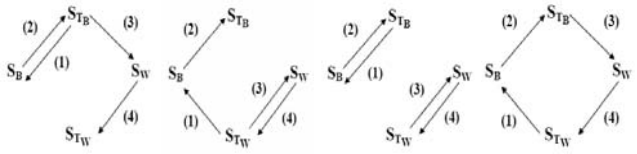
[2] Note that b_j (w_j) is used to denote the blob (word) and $\vec{b}_j$ ($\vec{w}_j$) is used to denote its corresponding vector representation.

[3] Similar matrix definition is widely used in *collaborative filtering* for modelling the user-item relation.

[4] Here the similarity function is in the form of inner-product, but note that our basic idea can also be applied to other kinds of similarity functions.

Table 1: Similarity calculation of different objects.

Object	Similarity Calculation		Intuition
Blob	$sim(b_i, b_j) = \vec{b}_i^\top \vec{b}_j = \Sigma_{k=1}^m \#(b_i, t_k)\#(b_j, t_k)$	$\mathbf{S_B} = \mathbf{M}\mathbf{M}^\top$	Similar Blobs: Appearing in the *same* images
	$sim(b_i, b_j) = \vec{b}_i^\top \mathbf{S_T} \vec{b}_j = \Sigma_{k_1=1}^m \Sigma_{k_2=1}^m \#(b_i, t_{k_1}) sim(t_{k_1}, t_{k_2}) \#(b_j, t_{k_2})$	$\mathbf{S_B} = \mathbf{M}\mathbf{S_T}\mathbf{M}^\top$	Similar Blobs: Appearing in the *similar* images
Image	$sim(t_i, t_j) = \vec{x}_i^\top \vec{x}_j = \Sigma_{k=1}^c \#(b_k, t_i)\#(b_k, t_j)$	$\mathbf{S_T} = \mathbf{M}^\top\mathbf{M}$	Similar Images: Sharing the *same* blobs
	$sim(t_i, t_j) = \vec{x}_i^\top \mathbf{S_B} \vec{x}_j = \Sigma_{k_1=1}^c \Sigma_{k_2=1}^c \#(b_{k_1}, t_i) sim(b_{k_1}, b_{k_2}) \#(b_{k_2}, t_j)$	$\mathbf{S_T} = \mathbf{M}^\top\mathbf{S_B}\mathbf{M}$	Similar Images: Sharing *similar* blobs
Word	$sim(w_i, w_j) = \vec{w}_i^\top \vec{w}_j = \Sigma_{k=1}^m \#(w_i, t_k)\#(w_j, t_k)$	$\mathbf{S_W} = \mathbf{N}\mathbf{N}^\top$	Similar Words: Appearing with the *same* images
	$sim(w_i, w_j) = \vec{w}_i^\top \mathbf{S_T} \vec{w}_j = \Sigma_{k1=1}^m \Sigma_{k2=1}^m \#(w_i, t_{k_1}) sim(t_{k_1}, t_{k_2}) \#(w_j, t_{k_2})$	$\mathbf{S_W} = \mathbf{N}\mathbf{S_T}\mathbf{N}^\top$	Similar Words: Appearing with *similar* images
Image	$sim(t_i, t_j) = \vec{y}_i^\top \vec{y}_j = \Sigma_{k=1}^d \#(w_k, t_i)\#(w_k, t_j)$	$\mathbf{S_T} = \mathbf{N}^\top\mathbf{N}$	Similar Images: Sharing the *same* words
	$sim(t_i, t_j) = \vec{y}_i^\top \mathbf{S_W} \vec{y}_j = \Sigma_{k_1=1}^d \Sigma_{k_2=1}^d \#(w_{k_1}, t_i) sim(w_{k_1}, w_{k_2}) \#(w_{k_2}, t_j)$	$\mathbf{S_T} = \mathbf{N}^\top\mathbf{S_W}\mathbf{N}$	Similar Images: Sharing *similar* words



(a) Type I *(b) Type II* *(c) Type III* *(d) Type IV*

Figure 1: Four types of interactions.

the similarity of images calculated based on their constituent blobs (as shown in Row 3 and 4 of Table 1) and their annotation words (as shown in Row 7 and 8 of Table 1), we denote the former as $\mathbf{S_{T_B}}$ and the latter as $\mathbf{S_{T_W}}$. The way of substituting $\mathbf{S_T}$ in Table 1 with $\mathbf{S_{T_B}}$ and $\mathbf{S_{T_W}}$ defines the following four types of interactions between blobs, images and words, as shown in Figure 1:

- **Type I:** $\mathbf{S_B} = \mathbf{M}\mathbf{S_{T_B}}\mathbf{M}^\top,\ \mathbf{S_W} = \mathbf{N}\mathbf{S_{T_B}}\mathbf{N}^\top$
  The interaction is restricted to images and blobs, but its influence can propagate to words via images.
- **Type II:** $\mathbf{S_B} = \mathbf{M}\mathbf{S_{T_W}}\mathbf{M}^\top,\ \mathbf{S_W} = \mathbf{N}\mathbf{S_{T_W}}\mathbf{N}^\top$
  The interaction is restricted to images and words, but its influence can propagate to blobs via images.
- **Type III:** $\mathbf{S_B} = \mathbf{M}\mathbf{S_{T_B}}\mathbf{M}^\top,\ \mathbf{S_W} = \mathbf{N}\mathbf{S_{T_W}}\mathbf{N}^\top$
  Blobs and words interact with images separately, and no information flows between blobs and words.
- **Type IV:** $\mathbf{S_B} = \mathbf{M}\mathbf{S_{T_W}}\mathbf{M}^\top,\ \mathbf{S_W} = \mathbf{N}\mathbf{S_{T_B}}\mathbf{N}^\top$
  There exist complete interactions between blobs, images and words, and the information can flow among all types of objects.

Following the directed edges in Figure 1, we formalize the iterative similarity updates with respect to each type of interactions, where the similarity can be decomposed into the initial similarity without any interaction with other types of objects and the refined similarity from other objects based on the interactions between objects.

- Type I:

$$\mathbf{S_B}^{(n)} = (1-\lambda_1)\mathbf{S_B}^{(0)} + \lambda_1 \mathbf{M}\mathbf{S_{T_B}}^{(n-1)}\mathbf{M}^\top \quad (1a)$$
$$\mathbf{S_{T_B}}^{(n)} = (1-\lambda_2)\mathbf{S_{T_B}}^{(0)} + \lambda_2 \mathbf{M}^\top\mathbf{S_B}^{(n-1)}\mathbf{M} \quad (1b)$$
$$\mathbf{S_W}^{(n)} = (1-\lambda_3)\mathbf{S_W}^{(0)} + \lambda_3 \mathbf{N}\mathbf{S_{T_B}}^{(n-1)}\mathbf{N}^\top \quad (1c)$$
$$\mathbf{S_{T_W}}^{(n)} = (1-\lambda_4)\mathbf{S_{T_W}}^{(0)} + \lambda_4 \mathbf{N}^\top\mathbf{S_W}^{(n-1)}\mathbf{N} \quad (1d)$$

- Type II:

$$\mathbf{S_B}^{(n)} = (1-\lambda_1)\mathbf{S_B}^{(0)} + \lambda_1 \mathbf{M}\mathbf{S_{T_W}}^{(n-1)}\mathbf{M}^\top \quad (2a)$$
$$\mathbf{S_{T_B}}^{(n)} = (1-\lambda_2)\mathbf{S_{T_B}}^{(0)} + \lambda_2 \mathbf{M}^\top\mathbf{S_B}^{(n-1)}\mathbf{M} \quad (2b)$$
$$\mathbf{S_W}^{(n)} = (1-\lambda_3)\mathbf{S_W}^{(0)} + \lambda_3 \mathbf{N}\mathbf{S_{T_W}}^{(n-1)}\mathbf{N}^\top \quad (2c)$$
$$\mathbf{S_{T_W}}^{(n)} = (1-\lambda_4)\mathbf{S_{T_W}}^{(0)} + \lambda_4 \mathbf{N}^\top\mathbf{S_W}^{(n-1)}\mathbf{N} \quad (2d)$$

- Type III:

$$\mathbf{S_B}^{(n)} = (1-\lambda_1)\mathbf{S_B}^{(0)} + \lambda_1 \mathbf{M}\mathbf{S_{T_B}}^{(n-1)}\mathbf{M}^\top \quad (3a)$$
$$\mathbf{S_{T_B}}^{(n)} = (1-\lambda_2)\mathbf{S_{T_B}}^{(0)} + \lambda_2 \mathbf{M}^\top\mathbf{S_B}^{(n-1)}\mathbf{M} \quad (3b)$$
$$\mathbf{S_W}^{(n)} = (1-\lambda_3)\mathbf{S_W}^{(0)} + \lambda_3 \mathbf{N}\mathbf{S_{T_B}}^{(n-1)}\mathbf{N}^\top \quad (3c)$$
$$\mathbf{S_{T_W}}^{(n)} = (1-\lambda_4)\mathbf{S_{T_W}}^{(0)} + \lambda_4 \mathbf{N}^\top\mathbf{S_W}^{(n-1)}\mathbf{N} \quad (3d)$$

- Type IV:

$$\mathbf{S_B}^{(n)} = (1-\lambda_1)\mathbf{S_B}^{(0)} + \lambda_1 \mathbf{M}\mathbf{S_{T_W}}^{(n-1)}\mathbf{M}^\top \quad (4a)$$
$$\mathbf{S_{T_B}}^{(n)} = (1-\lambda_2)\mathbf{S_{T_B}}^{(0)} + \lambda_2 \mathbf{M}^\top\mathbf{S_B}^{(n-1)}\mathbf{M} \quad (4b)$$
$$\mathbf{S_W}^{(n)} = (1-\lambda_3)\mathbf{S_W}^{(0)} + \lambda_3 \mathbf{N}\mathbf{S_{T_B}}^{(n-1)}\mathbf{N}^\top \quad (4c)$$
$$\mathbf{S_{T_W}}^{(n)} = (1-\lambda_4)\mathbf{S_{T_W}}^{(0)} + \lambda_4 \mathbf{N}^\top\mathbf{S_W}^{(n-1)}\mathbf{N} \quad (4d)$$

Here, $0 < \lambda_1, \lambda_2, \lambda_3, \lambda_4 < 1$, which balance the contribution of the refined similarities and the initial similarities $\mathbf{S_B}^{(0)} = \mathbf{M}\mathbf{M}^\top, \mathbf{S_{T_B}}^{(0)} = \mathbf{M}^\top\mathbf{M}, \mathbf{S_W}^{(0)} = \mathbf{N}\mathbf{N}^\top, \mathbf{S_{T_W}}^{(0)} = \mathbf{N}^\top\mathbf{N}$.

One question arises: Do the similarity refinement converge? In the following, we prove the convergence of the refinement of $\mathbf{S_B}, \mathbf{S_W}, \mathbf{S_{T_B}}$ and $\mathbf{S_{T_W}}$. Since Type IV interaction is the most complicated one among the four interactions, we provide the convergence analysis of $\mathbf{S_B}$ for Type IV as an example. The result is summarized in the following proposition. Similar results can be obtained for the other similarities for different types of interactions, and will be presented in a longer version.

Proposition 1. *Let* $\mathbf{C} = (\lambda_1\lambda_2\lambda_3\lambda_4)^{\frac{1}{2}}\mathbf{M}\mathbf{N}^\top\mathbf{N}\mathbf{M}^\top$, *whose eigenvalues are* $\{e_1, e_2, \cdots, e_c\}$. *If* $|e_i| < 1$ *(*$i = 1, \cdots, c$*), then* $\lim_{n\to\infty}\left(\mathbf{S_B}^{(n)} - \mathbf{S_B}^{(n-1)}\right) = 0$.

Proof.

$$\begin{aligned}
&\mathbf{S}_{\mathbf{B}}^{(n)} - \mathbf{S}_{\mathbf{B}}^{(n-1)} \\
&= \lambda_1 \mathbf{M}\left(\mathbf{S}_{\mathbf{T_W}}^{(n-1)} - \mathbf{S}_{\mathbf{T_W}}^{(n-2)}\right)\mathbf{M}^\top \\
&= \lambda_1\lambda_4 \mathbf{M}\mathbf{N}^\top\left(\mathbf{S}_{\mathbf{W}}^{(n-2)} - \mathbf{S}_{\mathbf{W}}^{(n-3)}\right)\mathbf{N}\mathbf{M}^\top \\
&= \lambda_1\lambda_4\lambda_3 \mathbf{M}\mathbf{N}^\top\mathbf{N}\left(\mathbf{S}_{\mathbf{T_B}}^{(n-3)} - \mathbf{S}_{\mathbf{T_B}}^{(n-4)}\right)\mathbf{N}^\top\mathbf{N}\mathbf{M}^\top \\
&= \lambda_1\lambda_4\lambda_3\lambda_2 \mathbf{M}\mathbf{N}^\top\mathbf{N}\mathbf{M}^\top\left(\mathbf{S}_{\mathbf{B}}^{(n-4)} - \mathbf{S}_{\mathbf{B}}^{(n-5)}\right)\mathbf{M}\mathbf{N}^\top\mathbf{N}\mathbf{M}^\top \\
&= \cdots = \mathbf{C}^p\left(\mathbf{S}_{\mathbf{B}}^{(q)} - \mathbf{S}_{\mathbf{B}}^{(q-1)}\right)\mathbf{C}^p
\end{aligned}$$

where $n = 4p + q$ ($q = 1, \cdots, 4$).

Since $\mathbf{C}$ is a symmetric matrix, which can be decomposed as $\mathbf{C} = \mathbf{Q}\mathbf{D}\mathbf{Q}^\top$, where $\mathbf{Q}$ is an orthogonal matrix and $\mathbf{D}$ is a diagonal matrix whose diagonal elements are the eigenvalues of $\mathbf{C}$. Substituting $\mathbf{C}$ in the above equation yields

$$\mathbf{S}_{\mathbf{B}}^{(n)} - \mathbf{S}_{\mathbf{B}}^{(n-1)} = \mathbf{Q}\mathbf{D}^p\mathbf{Q}^\top\left(\mathbf{S}_{\mathbf{B}}^{(q)} - \mathbf{S}_{\mathbf{B}}^{(q-1)}\right)\mathbf{Q}\mathbf{D}^p\mathbf{Q}^\top .$$

Since $|e_i| < 1$ ($i = 1, \cdots, c$), $\lim_{p\to\infty} \mathbf{D}^p = 0$. Therefore, we have $\lim_{n\to\infty}(\mathbf{S}_{\mathbf{B}}^{(n)} - \mathbf{S}_{\mathbf{B}}^{(n-1)}) = 0$. □

Based on the convergence of the similarities, we further derive the close-form solution of all the similarities for each interaction type. Due to space limitation, we only provide the derivation of $\mathbf{S_B}$ for Type IV and the other derivation will be presented in a longer version.

By plugging Eqs.(4b)∼(4d) into Eq.(4a) in Type IV, Eq.(4a) can be rewritten as

$$\mathbf{S}_{\mathbf{B}}^{(n)} = \mathbf{R} + \mathbf{C}\mathbf{S}_{\mathbf{B}}^{(n-4)}\mathbf{C} \tag{5}$$

where

$$\begin{aligned}
\mathbf{R} = &(1-\lambda_1)\mathbf{S}_{\mathbf{B}}^{(0)} + \lambda_1(1-\lambda_4)\mathbf{M}\mathbf{S}_{\mathbf{T_W}}^{(0)}\mathbf{M}^\top + \\
&\lambda_1\lambda_4(1-\lambda_3)\mathbf{M}\mathbf{N}^\top\mathbf{S}_{\mathbf{W}}^{(0)}\mathbf{N}\mathbf{M}^\top + \\
&\lambda_1\lambda_4\lambda_3(1-\lambda_2)\mathbf{M}\mathbf{N}^\top\mathbf{N}\mathbf{S}_{\mathbf{T_B}}^{(0)}\mathbf{N}^\top\mathbf{N}\mathbf{M}^\top
\end{aligned} \tag{6}$$

If $|e_i| < 1$ ($i = 1, \cdots, c$), by Proposition 1, assume that the sequence $\mathbf{S}_{\mathbf{B}}^{(n)}$ converges to $\mathbf{S}_{\mathbf{B}}^{*}$. When $n \to \infty$, since $\mathbf{C} = \mathbf{Q}\mathbf{D}\mathbf{Q}^\top$ and $\mathbf{D} = diag(e_1, e_2, \cdots, e_c)$, we obtain

$$\mathbf{S}_{\mathbf{B}}^{(*)} - \mathbf{Q}\mathbf{D}\mathbf{Q}^\top\mathbf{S}_{\mathbf{B}}^{(*)}\mathbf{Q}\mathbf{D}\mathbf{Q}^\top = \mathbf{R} . \tag{7}$$

With some algebra, we obtain the solution for Eq. 7 as:

$$\mathbf{S}_{\mathbf{B}}^{(*)} = \mathbf{Q}\left(\left(\mathbf{Q}^\top\mathbf{R}\mathbf{Q}\right) \odot \mathbf{\Sigma}\right)\mathbf{Q}^\top \tag{8}$$

where $\mathbf{\Sigma} = [\sigma_{ij}]_{c\times c}$, $\sigma_{ij} = (1 - e_i e_j)^{-1}$; $\odot$ is element-wise matrix multiplication.

Note that in many real tasks, data can be normalized before the similarity learning process to ensure the convergence.

3.2 Tackling the Four Tasks with Refined Similarities

We address how to use the refined $\mathbf{S_B}$ and $\mathbf{S_W}$ after exploiting multi-modal interactions to tackle the four image-text tasks as we mentioned in Section 1[5].

[5]Since these four tasks mainly concern about untagged images, the similarity matrices of tagged images, i.e., $\mathbf{S_{T_B}}$ and $\mathbf{S_{T_W}}$, are not used, but these matrices are obviously very useful for other tasks such as clustering tagged images.

Table 2: Simple methods to tackle the four tasks with the improved similarities.

Task:	**Content-based Image Retrieval**
Input:	Image Query q_{img}
Output:	A rank of u_i in U
Process:	a) For each $u_i = (\vec{z}_i, \phi)$ in U, $score_i = getScore(\vec{q}_{img}, \vec{z}_i, \mathbf{S_B})$ b) Sort $score_i$ decreasingly to get the rank of u_i
Task:	**Image Annotation**
Input:	Image Query q_{img}, the number of nearest neighbors k
Output:	A rank of w_i in W
Process:	a) For each $t_i = (\vec{x}_i, \vec{y}_i)$ in T, $score_i = getScore(\vec{q}_{img}, \vec{x}_i, \mathbf{S_B})$ b) Sort $score_i$ decreasingly to get a rank of t_i. If t_i belongs to the top k of this rank, set $\delta_i = \frac{1}{k}$;otherwise, set $\delta_i = 0$. c) $\vec{h} = \Sigma_{i=1}^{m}\delta_i\vec{y}_i$ d) Sort the element of $\vec{h}$ decreasingly to get the rank of w_i.
Task:	**Text-based Image Retrieval**
Input:	Text Query q_{txt}, the number of nearest neighbors k
Output:	A rank of u_i in U
Process:	a) For each $u_i = (\vec{z}_i, \phi)$ in U, repeat steps a)-c) of **Image Annotation** to get corresponding $\vec{h}_i$, $score_i = getScore(\vec{q}_{txt}, \vec{h}_i, \mathbf{I})$ % **I** *is identity matrix* b) Sort $score_i$ decreasingly to get the rank of u_i
Task:	**Query Expansion**
Input:	Text Query q_{txt}
Output:	A rank of w_i in W
Process:	a) For each w_i in W, construct $\vec{w}_i$ as a $d \times 1$ vector with the ith position as 1 and the other positions as 0, $score_i = getScore(\vec{q}_{txt}, \vec{w}_i, \mathbf{S_W})$ b) Sort $score_i$ decreasingly to get the rank of w_i

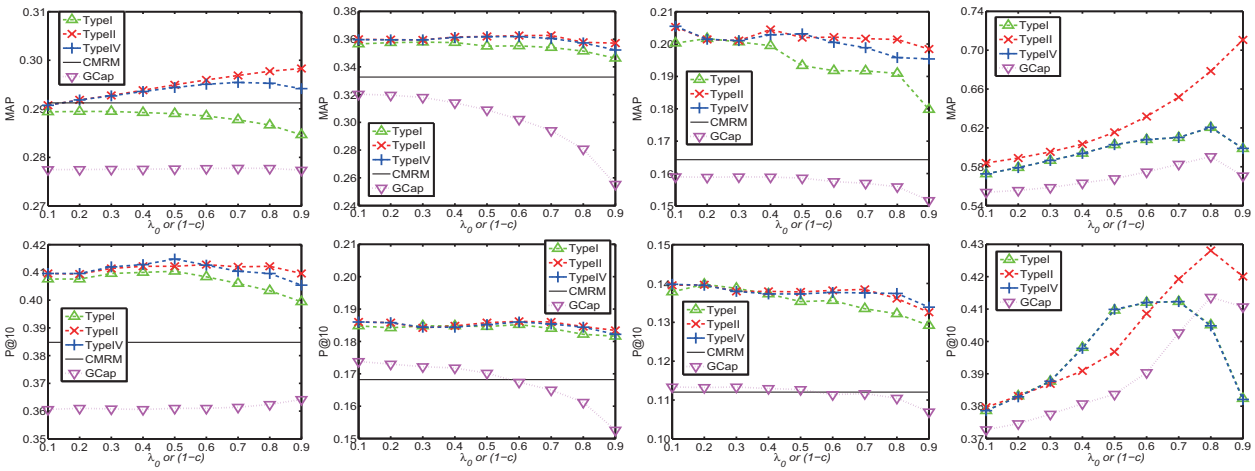
For content-based image retrieval and image annotation, the input is an image query q_{img} whose vector representation is $\vec{q}_{img} = [\ \#(b_1, q_{img}), \#(b_2, q_{img}), \cdots, \#(b_c, q_{img})]^\top$. For text-based image retrieval and query expansion, the input is a text query q_{txt} whose vector representation is $\vec{q}_{txt} = [\#(w_1, q_{txt}), \#(w_2, q_{txt}), \cdots, \#(w_d, q_{txt})]^\top$. We apply simple methods shown in Table 2 to solve these tasks based on the refined similarities $\mathbf{S_B}$ and $\mathbf{S_W}$.

In Table 2, the function $getScore(\vec{v}_1, \vec{v}_2, \mathbf{S})$ first uses $\mathbf{S}$ to obtain the "expanded" representations of $\vec{v}_1$, $\vec{v}_2$, where $\vec{v}_1' = \vec{v}_1 \times \mathbf{S}$ and $\vec{v}_2' = \vec{v}_2 \times \mathbf{S}$, and then, computes the score based on certain similarity measurement between $\vec{v}_1'$ and $\vec{v}_2'$.

4 Experiments

We use the database in [Duygulu *et al.*, 2002] in our experiments. This database contains 5,000 images from 50 COREL Stock Photo CDs, each of which contains 100 images on a topic. 499 image blobs are generated through image segmentation using normalized cuts and region clustering using k-means. The number of blobs contained in each image ranges from 1 to 10. 374 words are used to annotate the images,and the number of words in each image ranges from 1 to 5. The training set consists of 4,500 images and the test set consists of 500 images. We use the training set as the tagged images and regard the test set as untagged images.

As shown in Table 2, the outputs of the four tasks are all ranks. Thus, we use the standard non-interpolated mean average precision (MAP) and precision at 10 (P@10) to measure the performance. MAP measures the precision of the whole ranking list and P@10 reflects the precision of the top 10 elements in the ranking list. Details can be found in [Baeza-Yates and Ribeiro-Neto, 1999].



(a) CBIR (b) Image Annotation (c) TBIR (d) Query Expansion

Figure 2: The influence of λ_0

First, we focus on *content-based image retrieval*, *image annotation* and *text-based image retrieval*. In our experiments, we compare the performance of using different values of $\mathbf{S_B}$ for these three tasks. The 'Baseline' method does not consider the relations between blobs and assume two images are similar when they contain the same blobs, thus the identity matrix is used for $\mathbf{S_B}$, i.e., $\mathbf{S_B} = \mathbf{I}$. The 'Initial' method uses the initial value of $\mathbf{S_B}$, i.e., $\mathbf{S_B} = \mathbf{S_B}^{(0)} = \mathbf{M}\mathbf{M}^\top$. The 'Type I', 'Type II' and 'Type IV' methods use the final similarity generated by the interactions of Type I, Type II and Type IV [6], i.e., $\mathbf{S_B} = \mathbf{S_B}^{(*)}$. By comparing 'Baseline' with 'Initial' and 'Type I/II/IV', we can show the effect of considering the relations between blobs.

For 'Type I/II/IV', the parameters λ_1 to λ_4 are simply set to the same value, denoted by λ_0. We set $\lambda_0 = 0.5$, which implies that for any type of objects, its own similarity and the influence from other types of objects are equally important. Other values of λ_0 will be studied at the end of this section. For 'Baseline', 'Initial' and 'Type I/II/IV', when they are used to solve *image annotation* and *text-based image retrieval* tasks, the number of nearest neighbors k is set as 100 using the validation set (see Table 2). To avoid decomposing a large matrix, we normalize the matrices into probability transition matrices and conduct random walk solution iteratively[7].

CMRM, CLP and SVM are the state-of-the-art techniques for *image annotation*. Using the strategy of Datta *et al.* [2006], these methods can also be applied to *content-based image annotation* and *text-based image annotation*. GCap can be used for all these three tasks. For CMRM, the parameters α and β are set to 0.1 and 0.9, respectively, which are the best parameters tuned on validation set. For CLP, the parameter β is set to 0.9 based on the validation set. For SVM, we use LibSVM with default parameters. For GCap, the parameter c which balances the probability of random walking and restarting is very similar to λ_0 of our method. c is first selected as 0.7 according to Pan *et al.* [2004]'s suggestion and the influence of c is studied with λ_0 at the end of this section. In general, the performance of these state-of-the-art techniques provides a good baseline.

Table 3: Comparison of MAP and P@10 on different tasks. (TBIR: Text-Based Image Retrieval; CBIR: Content-Based Image Retrieval; the best performance of each column is bolded.)

Tasks:	CBIR		Img Annotation		TBIR		Query Expansion	
	MAP	P@10	MAP	P@10	MAP	P@10	MAP	P@10
Baseline	0.274	0.365	0.320	0.176	0.154	0.109	N/A	N/A
Initial	0.289	0.408	0.356	0.185	0.197	**0.138**	0.567	0.373
Type I	0.289	0.410	0.355	0.185	0.193	0.135	0.603	**0.410**
Type II	**0.295**	0.412	**0.362**	**0.186**	0.202	**0.138**	**0.615**	0.397
Type IV	0.294	**0.415**	**0.362**	0.185	**0.203**	0.137	0.603	**0.410**
CMRM	0.291	0.385	0.333	0.168	0.164	0.112	N/A	N/A
CLP	0.287	0.381	0.302	0.164	0.164	0.112	N/A	N/A
SVM	0.284	0.397	0.305	0.161	0.139	0.097	N/A	N/A
GCap	0.278	0.361	0.318	0.172	0.159	0.113	0.559	0.378

In the experiments on *content-based image retrieval*, queries consisting of 500 images in the test set and images sharing some annotation words with the query are regarded as relevant images. In the experiments on *image annotation*, for the returned rank of annotation words, a word is regarded as relevant if it is in the annotation of the query image. We also use 500 images in the test set as queries. For *text-based image retrieval*, an image is regarded as relevant to the text query if its annotation contains all the query words. The query set consists of 179 one-word queries, 385 two-word queries, 176 three-word queries and 24 four-word queries. These queries appear at least two times in the annotations in imagebase. The performances of our methods and the state-of-the-art techniques for these three tasks are reported in Table 3.

Table 3 shows that for these three tasks, 'Type I/II/IV' and 'Initial' consistently and significantly outperform the other compared methods. Such an observation indicates that considering the relation between blobs is beneficial. Compar-

[6] In Type III, $\mathbf{S_B}$ is the same as Type I and $\mathbf{S_W}$ is the same as Type II, thus we do not report the performance of Type III.

[7] This iterative approach converges quickly. For example, for Type I interaction, it converges in no more than 10 iterations.

ing 'Type I/II/IV' and 'Initial', we can find that exploiting the interactions between different types of objects can further improve the performance achieved by exploiting the relation between blobs. Such superiority is obvious in terms of MAP rather than P@10, which suggests that exploiting multimodal interactions can help improve the ranks of some "hard-to-identified" relevant items and hence result in a better overall ranking list. Comparing three interaction types, we can observe 'Type II/IV' consistently perform better than 'Type I'. Recall that in 'Type I', information never flows back from "words" to the other interacting types. Since information in blobs might be less reliable than words due to the semantic gap, it is natural for 'Type II/IV' to be superior.

Second, we study *query expansion*. This task requires the involvement of the user, thus it is difficult to measure the performance of this task directly. Instead, we attempt to measure $\mathbf{S_W}$, which is essential for query expansion. As mentioned above, the experimental database contains 50 topics. The correlation of words based on their distributions over these 50 topics is used to assign each word a ground truth rank of the other words according to their relatedness to the concerned word. Note that this topic information is only used here for evaluation purpose and is not used for any other step. Here, we simply regard the top 5 words of each word's ground truth rank as relevant and the others are irrelevant. Then, we measure how good the word-to-word relations reflected by $\mathbf{S_W}$ are. The 'Baseline' method could not be applied here, since $\mathbf{S_W} = \mathbf{I}$ is not helpful for query expansion. The 'Initial' method uses $\mathbf{S_W} = \mathbf{N}\mathbf{N}^\top$. The 'Type I/II/IV' methods use $\mathbf{S_W} = \mathbf{S_W}^{(*)}$ generated by the interactions of Type I/II/IV. Only GCap is evaluated here, since other compared methods are not applicable to query expansion. The average performance is reported in Table 3. Similar to the other three tasks, for query expansion, every interaction type performs significantly better than 'Initial' and GCap.

In the experiments, λ_0 is fixed to 0.5. Now we study the influence of different λ_0 on the performance of the proposed method. The performance of GCap with different c is also reported for comparison. CMRM is used here for reference. The results are provided in Figure 2. It is obvious that with different λ_0, our proposed method almost always outperforms CMRM and GCap, except that when $\lambda_0 \geq 0.8$, GCap performs better than 'Type I/IV' in terms of P@10. Note that 'Type I/IV' still perform better than GCap in terms of MAP. It suggests that even if 'Type I/IV' might put fewer relevant words in the first 10 words of the ranking list, the ranks assigned to the relevant words can still be higher than that assigned by GCap to yields better MAP.

5 Conclusion

Given an imagebase which contains tagged images, four types of tasks can be executed, i.e., content-based image retrieval, image annotation, text-based image retrieval, and query expansion. For any of these tasks the similarity on the concerned type of objects is essential. Usually three major types of objects in an imagebase are image blobs, annotation words and images. Since these objects have strong interactions, it is reasonable to refine the similarity of one type of objects with the help of that of another type of objects.

In this paper, we propose a unified framework to model such interactions. Experiments show that, with the similarity refined by the proposed method, simple methods can outperform the state-of-the-art methods.

Currently we represent each image as a group of blobs. Extending our proposal to other representations is an interesting issue for future work. Automatically selecting the parameters λ_1-λ_4 for different types of interactions is another interesting future work.

References

[Baeza-Yates and Ribeiro-Neto, 1999] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison-Wesley, Wokingham, UK, 1999.

[Barnard *et al.*, 2003] K. Barnard, P. Duygulu, D. Forsyth, N. de Freitas, D. M. Blei, and M. J. Jordan. Matching words and pictures. *JMLR*, 3:1107–1135, 2003.

[Blei and Jordan, 2003] D. M. Blei and M. I. Jordan. Modeling annotated data. In *SIGIR*, pages 127–134, 2003.

[Datta *et al.*, 2006] R. Datta, W. Ge, J. Li, and J. Z. Wang. Toward bridging the annotation-retrieval gap in image search by a generative modeling approach. In *ACMMM*, pages 977–986, 2006.

[Duygulu *et al.*, 2002] P. Duygulu, K. Barnard, N. de Freitas, and D. Forsyth. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *ECCV*, pages 97–112, 2002.

[Guo *et al.*, 2007] Z. Guo, Z. Zhang, E. P. Xing, and C. Faloutsos. Enhanced max margin learning on multimodal data mining in a multimedia database. In *KDD*, pages 340–349, 2007.

[Hardoon *et al.*, 2003] D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor. Canonical correlation analysis: An overview with application to learning methods. Technical Report CSD-TR-03-02, University of London, 2003.

[Jeon *et al.*, 2003] J. Jeon, V. Lavrenko, and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. In *SIGIR*, pages 119–126, 2003.

[Kandola *et al.*, 2003] J. Kandola, J. Shawe-Taylor, and N. Cristianini. Learning semantic similarity. In *NIPS 15*, pages 657–664. 2003.

[Kang *et al.*, 2006] F. Kang, R. Jin, and R. Sukthankar. Correlated label propagation with application to multi-label learning. In *CVPR*, pages 1719–1726, 2006.

[Lafferty and Zhai, 2001] J. Lafferty and C. Zhai. Document language models, query models, and risk minimization for information retrieval. In *SIGIR*, pages 111–119, 2001.

[Li and Wang, 2003] J. Li and J. Z. Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *IEEE TPAMI*, 25(9):1075–1088, 2003.

[Pan *et al.*, 2004] J.-Y. Pan, H.-J. Yang, C. Faloutsos, and P. Duygulu. GCap: Graph-based auotmatic image captioning. In *MDDE*, pages 146–156, 2004.

[Smeulders *et al.*, 2000] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *IEEE TPAMI*, 22(12):1349–1380, 2000.

[Xing *et al.*, 2005] E. P. Xing, R. Yan, and A. G. Hauptmann. Mining associated text and images with dual-wing harmoniums. In *UAI*, pages 633–641, 2005.