Tensor product of correlated text and visual features: a quantum theory inspired image retrieval framework

Conference or Workshop Item

Tensor Product of Correlated Text and Visual Features: A Quantum Theory Inspired Image Retrieval Framework

Jun Wang and **Dawei Song** and **Leszek Kaliciak**
School of Computing, The Robert Gordon University, Aberdeen, UK
{j.wang3,d.song,l.kaliciak}@rgu.ac.uk

Abstract

In multimedia information retrieval, where a document may contain textual and visual content features, the ranking of documents is often computed by heuristically combining the feature spaces of different media types or combining the ranking scores computed independently from different feature spaces. In this paper, we propose a principled approach inspired by Quantum Theory. Specifically, we propose a tensor product based model aiming to represent text and visual content features of an image as a non-separable composite system. The ranking scores of the images are then computed in the form of a quantum measurement. In addition, the correlations between features of different media types are incorporated in the framework. Experiments on ImageClef2007 show a promising performance of the tensor based approach.

Introduction

With rapidly increasing volume of digital image data, e.g. in specialised image repositories, social photo sharing sites and all sorts of multimedia documents on the Web, an effective search for images that satisfy users' information needs is becoming a challenging research topic.

In the early stage of image retrieval research, librarians had to attach some keywords to each image in order to retrieve relevant images with text retrieval techniques. Nowadays, however, manual labelling becomes infeasible due to the increasing size of the image collections. To circumvent such obstacle, content-based image retrieval (CBIR), which uses visual features to measure the content similarity between images, has been investigated. Typical visual features include colour histogram, texture and shape, etc. An image is represented as a vector in a feature space. For example, each dimension in a colour histogram space corresponds to a color bin along channels R-G-B or H-S-V, and the value of an image on each dimension is the normalized number of pixels in the image falling into the corresponding bin. The similarity between two images can be measured based on how close their corresponding vectors are on the feature space, e.g. through the Cosine function. Nevertheless, even the start-of-art CBIR techniques can only achieve a limited performance because of the semantic gap between the content and its high level semantics. Given that more and more images and multimedia documents contain both visual content and certain amount of text annotations (e.g. tags, metadata, text descriptions, etc.), combining the textual and visual features of images for image retrieval has recently attracted increasing attention.

Three commonly adopted combination methods are: 1) using textual data to retrieve images, then re-ranking the retrieval results with their visual feature (Yanai 2003); 2) or using visual feature to retrieve images, then re-ranking the results with their textual features (Tjondronegoro et al. 2005); 3) or combining linearly the feature spaces or the similarity scores based on different features (Rahman, Bhattacharya, and Desai 2009)(Matthew Simpson 2009)(Min 2004). All these combination methods treat the textual and visual features of images individually, and combine them in a rather heuristic manner. Therefore it is difficult to capture the relationship between them. Indeed, as both the textual and visual features describe the same image, there are inherent correlations between them and they should be incorporated into the retrieval process as a whole in a more principled way.

In this paper, we present a Quantum Theory inspired retrieval model based on the tensor product of the textual and visual features. It describes an annotated image as a n-order tensor in order to catch the non-separability of textual and visual features. The order of the tensor depends on the visual features that are going to be incorporated in the image retrieval. Currently we focus on 2nd-order tensor.

In practice, not every image is associated with a proper textual annotation: some annotations do not describe the content of the image at all and some images do not even contain any textual information. Ideally, the problem can be alleviated by automatically annotating images with controlled textual labels, usually through supervised learning from pre-annotated training examples, at the pre-processing stage.

However, the automatic annotation is out of scope of this paper, and is an ongoing research topic on its own. Instead, in this paper, we are concerned about a finer-grained correlation between the dimensions across the textual and visual feature spaces. We present two rather straightforward statistical methods to associate dimensions (e.g. words) of the textual feature space with the dimensions (e.g. the HSV colour bins) of the visual feature space, while the main focus of the paper is to build and test the unified image retrieval framework.

This paper is structured as follows. We first show how to represent images with different media types using tensor product. We then present a model for how to score the images in the tensor space with a quantum-like measurement and how the correlations between dimensions across different feature spaces can be incorporated. Two statistical methods for deriving the cross-feature correlations are proposed. We then report our experimental results on ImageClef2007, a standard image retrieval benchmarking collection, to demonstrate the potential of the proposed model. Observations from the results are discussed, leading to our conclusions and future work.

Tensor Product

Tensors are geometric entities introduced into mathematics and physics to extend the notion of scalars, (geometric) vectors and matrices. Tensor can express the relationship of vector spaces.

The tensor product is used to construct a new vector space or a new tensor. The result of tensor product of two Hilbert spaces is another Hilbert space associated with a composite system, which is constructed by the two single systems from the two sub-spaces.

Given two Hilbert spaces A and B, their tensor product $A \otimes B$ can be defined in the following, where we assume that the dimensionality of the spaces is finite. Let $\{|a_i\rangle : i = 1...m\}$ be the orthogonal basis of space A, and $\{|b_j\rangle : j = 1...n\}$ be the orthogonal basis of space B. Then $|a_i\rangle \otimes |b_j\rangle$ constitutes the basis of the tensor space $A \otimes B$.

Suppose two single systems can be represented as: $|a\rangle = \sum_i \alpha_i |a_i\rangle$ and $|b\rangle = \sum_j \beta_j |b_j\rangle$. Then the composite system containing $|a\rangle$ and $|b\rangle$ in the tensor space can be expressed as:

$$|\phi\rangle = \sum_i \sum_j \gamma_{ij} |a_i\rangle \otimes |b_j\rangle \;\; or \;\; \sum_i \sum_j \gamma_{ij} |a_i b_j\rangle \qquad (1)$$

If each γ_{ij} can be decomposed as $\alpha_i \cdot \beta_j$, then $|\phi\rangle = |a\rangle \otimes |b\rangle$, which means that systems $|a\rangle$ and $|b\rangle$ are independent. Otherwise they are entangled or non-separable.

Next let us look at how to represent a multimedia document in the Hilbert space. Traditionally documents are represented as vectors. For example, when a document is represented by its textual feature, denoted $d^T = (tf_1, tf_2, \cdots, tf_n)^T$, where tf_n is the frequency that term t_n appears in the document d, and tf_n is zero when term t_n does not appear in d. The visual feature representation for the document can be in the same form, e.g. $d^F = (f_1, f_2, \cdots, f_m)^T$, where F denotes the type of visual feature and f_i refers to the feature value of ith-dimension in the feature space.

In each individual feature space, a document can be written as a superposition state. In text feature space, H_T: $|d\rangle_T = \sum_i w_{t_i} |t_i\rangle$, where $\sum_i w_{t_i}^2 = 1$. As the amplitude w_{t_i} for each state $|t_i\rangle$ should be proportion to the probability that the document is about the term t_i, it can be set as normalized term frequency of t_i, e.g. $w_{t_i} = tf_i / \sqrt{\sum_j^n tf_j^2}$. Note that the amplitude w_{t_i} can be set up with any other traditional term weighting scheme, e.g. TF-IDF. The only restriction here is to make sure that the sum of $w_{t_i}^2$ should be equal to one. Similarly a document can also be described as a superposition state in a content feature space H_F: $|d\rangle_F = \sum_i w_{f_i} |f_i\rangle$.

When to measure probability that a document is about a text or a visual feature, i.e. the probability that the superposition document collapses to a certain state, we can apply the vector product $P(t_i|d) = |\langle t_i|d\rangle_T|^2 = w_{t_i}^2$, which can be written as a projection to a space spanned by $|t_i\rangle$: $P(t_i|d) = \langle t_i|\rho_d|t_i\rangle = w_{t_i}^2$, where $\rho_d = |d\rangle\langle d|$ is the density matrix of document d. We will describe density matrix in more detail in the next section.

Naturally when we try to combine the textual and visual systems, we use the tensor product to get a unified system:

$$|d\rangle_{TF} = |d\rangle_T \otimes |d\rangle_F = \sum_{ij} \gamma_{ij} |t_i\rangle \otimes |f_j\rangle$$

The tensor space opens a door to linking and expanding the individual feature spaces as non-separable systems and allowing the correlations existing between them to be naturally incorporated in the unified theoretical framework.

When the visual feature and textual feature are independent, the amplitudes in this composite system can be written as i.e.$\gamma_{ij} = \alpha_i \cdot \beta_j$. However, this is not always the case in general. The features can be highly correlated in some states, i.e. some keywords may be highly correlated with some visual features. We will show later how we describe such correlations in the formal model.

With a superposed multimedia document, the density matrix of its sub-systems can be presented as $\rho_{d_T} = \sum_i \alpha_i^2 |t_i\rangle\langle t_i|$, and $\rho_{d_F} = \sum_i \beta_i^2 |f_i\rangle\langle f_i|$. When $|d\rangle_T$ and $|d\rangle_F$ are independent, the density matrix for the composite system is:

$$\rho_{d_{TF}} = \sum_{ij} \alpha_i^2 \beta_j^2 |t_i f_j\rangle \otimes \langle t_i f_j| = \rho_{d_T} \otimes \rho_{d_F} \qquad (2)$$

As mentioned before, in most situations, the two systems are not independent, i.e. $\rho_{d_{TF}} = \sum_{ij} \gamma_{ij}^2 |t_i f_j\rangle \otimes \langle t_i f_j|$. We can always separate the correlation term:

$$\rho_{d_{TF}} = \rho_{d_T} \otimes \rho_{d_F} + \rho_{correlation} \qquad (3)$$

How to compute the $\rho_{correlation}$ is a challenging open research question. Currently we use some simple statistical methods for the purpose, which will be introduced in the later sections.

Density Matrix and Measurement Operator

Density Operator: document

In quantum mechanics, a density matrix is a self-adjoint (or Hermitian) positive-semidefinite matrix, of trace one, which describes the statistical state of a quantum system.

1) Density matrix:

The formal density matrix definition is:

$$\rho = p_i|\phi_i\rangle\langle\phi_i| \tag{4}$$

Where p_i is the probability of the system being in the state $|\phi_i\rangle$, or the proportion of the ensembles being in the state $|\phi_i\rangle$. Density matrix can be used to describe both pure and mixed state system.

2.) Trace invariant:

The basis ϕ_i does not have to be orthogonal. If the basis for density matrix is not orthogonal, we can always change the basis.

$$\begin{aligned}\rho_d &= \sum_i w_i|\phi_i\rangle\langle\phi_i| &(5)\\ &= \sum_i w_i \sum_j U_{ij}|\varphi_j\rangle \sum_k \langle\varphi_k|U_{ik} &(6)\\ &= \sum_{jk} \rho_{jk}|\varphi_j\rangle\langle\varphi_k| &(7)\end{aligned}$$

where we have $w_i = \alpha_i^2$, $\rho_{jk} = \sum_i U_{ij}U_{ik}w_i$ and $tr(\rho_d) = 1$.

In this case the observed value is:

$$\begin{aligned}\langle A\rangle &= tr(\rho A) &(8)\\ &= \sum_{jk} \rho_{kj}A_{jk}\end{aligned}$$

3) Density matrix of a document:

Similarly, when preparing a document density matrix, we can assume each term as a state t_i. If we assume each term $|t_i\rangle$ is a orthonormal basis:

$$\langle t_i|t_j\rangle = \delta_{ij} \tag{9}$$

Then we do not need to change the basis, and the density matrix of the document is:

$$\rho_d = \sum_i w_i|t_i\rangle\langle t_i| \tag{10}$$

This density matrix is a diagonal matrix with trace 1, whose entry corresponds to the probability that the document is about the term t_i.

If we assume that $|t_i\rangle$ are not orthogonal to each other, then we can always represent the document with orthogonal base like what we have showed in Equation 7.

$$\begin{aligned}\rho_d &= \sum_i |t_i\rangle\langle t_i| &(11)\\ &= \sum_i w_i \sum_j U_{ij}|e_j\rangle \sum_k \langle e_k|U_{ik} &(12)\\ &= \sum_{jk} \rho_{jk}|e_j\rangle\langle e_k| &(13)\end{aligned}$$

Observable: query

In quantum physics, a system observable is a property of the system state that can be determined by some sequence of physical operations. The mean value over the observable O is:

$$\begin{aligned}\langle O\rangle &= \langle\phi|O|\phi\rangle &(14)\\ &= \sum_i c_i^2\lambda_i &(15)\end{aligned}$$

λ_i is the eigen value of observable O. As in quantum mechanic, only when an eigen value of the observable is measured, the system state can be postulated on which state it collapsed.

The concept of measurement in quantum theory fits the IR problem well: considering a query an observable, the higher similarity value measured on a document, the higher relevance of the document to the query.

The density of a query is:

$$O = \rho_q = \sum_i q_i^2|t_i\rangle\langle t_i| \tag{16}$$

If possible, we can use spectrum to represent the observable:

$$\begin{aligned}O &= \sum_i \lambda_i|e_i\rangle\langle e_i| &(17)\\ & &(18)\end{aligned}$$

$|e_i\rangle$ is the eigen basis of observable O, and λ_i is the corresponding eigen value.

Measurement on a Document

To gain the expected value of O, we need to prepare the document density matrix based on the eigen state of the observable:

$$\rho'_d = U\rho_d U' \tag{19}$$

ρ' and ρ define the same density matrix if and only if there is a unitary matrix U with $U'U = I$

$$|t'_i\rangle\sqrt{w'_i} = \sum_j U_{ij}|t_i\rangle\sqrt{w_i} \tag{20}$$

With quantum measurement :

$$\begin{aligned}\langle O\rangle &= tr(U\rho_d U'UOU') &(21)\\ &= tr(U\rho_d U'U\rho_q U') &(22)\end{aligned}$$

In this paper, we assume that $|t_i\rangle$ are orthogonal, then

$$\begin{aligned}\langle O\rangle &= tr(\sum_i \alpha_i^2|t_i\rangle\langle t_i|O) &(23)\\ &= tr(\sum_i \alpha_i^2|t_i\rangle\langle t_i| \sum_j q_j^2|t_j\rangle\langle t_j|) &(24)\\ &= \sum_i \alpha_i^2 q_i^2 &(25)\end{aligned}$$

Correlating Visual and Text Features

We now present how we identify the correlation between text and visual features, and embed them into density matrix of the composite system. Here, we are interested in the correlation between the individual dimensions from the two feature spaces: i.e., how some keywords in the text feature space are correlated with certain colour bins in the HSV feature space. Two statistical methods have been developed.

Maximum Feature Likehood

The assumption behind this method is: if some images are about same or similar thing, they tend to have similar visual features. For example, when an image contains sea, there is usually a large area in the image being blue. Thereafter its visual feature space should express as such.

When an image has no text information, we can check whether this image has a distinctive visual feature value in certain dimension. If the image has a very large value for that particular dimension f_j which correlate with sea, we can conjecture that it is very likely that sea appears in the image and would like to link it with the keyword "sea". In this way, an image initially without keyword "sea" can be retrieved by a text query about sea.

Operationally, to associate the text with the maximal likely visual feature dimension, we first group images by keywords to find a subset S_t of images containing word t in their annotations:

$$S_t = \{d_i | t \in title(d_i)\}. \tag{26}$$

Within this subset, the feature dimension on which most images have the highest feature values can be detected by choosing a feature dimension having the highest average feature value:

$$i = argmax_i(\{\bar{f_1}, \bar{f_2}, \cdots, \bar{f_n}\} \tag{27}$$

Here, n is the dimensionality of the visual feature space. $\bar{f_i}$ is the average feature value on the i-th feature dimension across the subset S_t. Thereafter, the dimension i of visual feature will be associated with text t.

Sometimes, however, the maximal value of a feature dimension is not distinctively greater than other dimensions, or the maximal value occurs due to the fact that many images have very high value in this dimension. To avoid such situation, only the feature dimensions whose average value is substantially (e.g., 2 standard deviation) higher than the average value on all the dimensions can be associated with the text. For example, for the subset S_t with average feature values $\{\bar{f_1}, \bar{f_2}, ..., \bar{f_n}\}$ and highest feature value dimension $x = argmax_i(\{\bar{f_i}\})$, the necessary condition to associate the x-th dimension with term t is $\bar{f_x} \geq average(\bar{f_i}) + 2std(\bar{f_i})$. Also across the subset, the feature value for this dimension $\{f_{1x}, f_{2x}, ..., f_{mx}\}$ should take a t-test to make sure that the high average feature value is not due to few images having very high feature value on this dimension.

It is also possible that some dimensions could be associated with too many words. To avoid this, we set up a threshold for each word to decide whether it needs to have a feature dimension association. Currently we choose the words occurring in more than 25 and less than 65 images. This is because if a term occurs in too few documents, the correlation between the term and visual feature could be a casual relation. While if a term occurs in too many documents, then this term may not be a good informative term, and will not help much in the retrieval even if these terms do associate with certain visual feature dimensions.

This method is straightforward and aims to solve the problem that some images do not have annotation. Nevertheless, we should bear in mind that such processing can bring the textual noise to the images. It can also miss some appropriate text which does not associate to the maximal feature value, but the moderate feature value. We observed that sometimes the textual feature does not necessarily associate with the highest visual feature value.

In the next step, we explore the word-feature co-occurrence method.

Feature and Word Mutual Information Matrix

To discover the visual and textual feature's semantic connection, we use the mutual information matrix. The mutual information of two random variables is a quantity that measures the mutual dependence of the two variables. Here we use it to describe the dependency between the visual and textual features. Each visual feature dimension is possible- to associate with each text with certain degree.

The entry of the matrix is the mutual information of feature on dimension f_i and text on t_i, which is defined as the follows:

$$FT_{ij} = MI(f_i, t_j) = \log_2 \frac{P(f_i, t_j)}{P(f_i)P(t_j)} \tag{28}$$

Here are the definitions of each probability:

- $P(f_i, t_j)$ is the the probability that a document contains word t_j and pixels in HSV bin i, $P(f_i, t_j) = \frac{N_{Pixel}(f_i, t_j, c)}{N_{Pixel}(c)}$.
- $P(t_j)$ is the probability that term t_j appears in the collection. We use geometric distribution $P(t_j) = 1 - (1-p)^k$ to represent the term distribution, which fits to the term occurrence distribution of our test collection. Here, p is a shape parameter and $p = 0.5$, k is the frequency of term t_j occurring in the collection.
- $P(f_i)$ is the probability that a pixel falls into feature bin i, $P(f_i) = \frac{N_{Pixel}(f_i, c)}{N_{Pixel}(c)}$.

The mutual information matrix is used to compute the association score between a term and a image, while based on the term's correlations with each visual feature dimension. When the mutual information is summed up with respect to a word t, according to an image's visual feature, an association score between the image and the word can be derived.

$$score(d, t) = \sum_i P(f_i|d) \cdot MI(f_i, t) \tag{29}$$

Suppose the document has a feature vector $F = (f_1, f_2, ...f_n)$, then the expected association score for each word will be $C = F \cdot FT$, which in turn can be used to build density correlation between textual and visual features $\rho_{correlation}$:

$$|d\rangle_T^{expand} = \sum_i c_i |t_i\rangle, \;\; C = F \cdot FT \tag{30}$$

$$\rho_{correlation}^d = \sum_{ij} c_i \beta_j |t_i f_j\rangle\langle t_i f_j| \tag{31}$$

In practice, we can only choose top n highly scored words to create the correlation density matrix, in order to reduce the computational cost. The setting of n varies from 5 to 30 in our experiment.

Experimental Settings and Results

Settings

The image retrieval experiments based on a tensor space are carried out on ImageClef2007, a widely used benchmarking collection for image retrieval. This collection has totally 20,000 images, each with an annotation file including fields such as title, description and note, etc. 60 test queries are provided, together with the ground truth data. Each query consists of 3 sample images and a text description.

Figure 1 shows an example image and its annotation file.

Figure 1: Image 112.jpg

```
<DOC>
<DOCNO>annotations/00/112.eng</DOCNO>
<TITLE>Excursion with the godchildren</TITLE>
<DESCRIPTION></DESCRIPTION>
<NOTES></NOTES>
<LOCATION>Quilotoa, Ecuador</LOCATION>
<DATE>April 2002</DATE>
<IMAGE>images/00/112.jpg</IMAGE>
<THUMBNAIL>thumbnails/00/112.jpg</THUMBNAIL>
</DOC>
```

In each annotation file, the informative texts mainly appear in the title and location fields. A small portion of images also have some short notes or descriptions. Therefore, in our experiments, the texts are extracted from “title”, “notes” and “location” fields.

When creating density matrix for each document, we set the probability of each term with the normalised TF-IDF. The visual feature that we are using here is HSV color histogram, which is the cylinder representation of RGB color space. In the cylinder, the angle around the central vertical axis corresponds to hue, the distance from the axis corresponds to saturation, and the distance along the axis corresponds to lightness or brightness. Hue and saturation components help to retain light independent color properties. The HSV color histogram of an image is computed as three independent distributions. Firstly, we split the image into individual color channels (grayscale representations of primary colors). Next, we discretize the colors and count how many pixels belong to each color bin. The mapping between color and bin index can be defined as $i = f(h, s, v)$.

Note that the feature space can be replaced by any other visual feature in the future. The tensor space can also be expanded with more visual feature space when needed.

Clearly, each dimension of the HSV color space is orthogonal to the others, as the color in one dimension (a HSV color bin) does not overlap with the color in another dimension. However, this is not true for textual feature space, in which the dimensions are words. Because some words can have same or similar meanings, e.g. mug and cup both count when people search for some tableware in the picture. Actually they may refer to the same thing most of the time. This can be sought by replacing the synonyms by one unique term, or using LSI to find its latent semantic space. As a first step, in this paper, we simply assume that the words are orthogonal, and focus on the tensor model itself.

Image Ranking

In our experiments, we compare the proposed tensor product based model with methods based on the pure visual and pure textural features individually. We also compare with the use of simple concatenation of textual and visual feature vectors. Because of the slight adjustment of each model during the experiments, we list the name of each individual run and its explanation as follows:

- cbF: pure visual feature based method using the city block distance measure (following the recommendation from a systematic study on distance measurements in (Liu et al. 2008)).
- cosT: pure text-based method using cosine similarity
- cos T+F: cosine similarity based on the concatenation of textual and visual features
- cosT(e)+F: cosine similarity based on the concatenation of textual and visual features (Each image will be annotated with some associated words first.)
- tensor(T+F): quantum-like measurement in tensor space
- tensorT(e)+F: quantum-like measurement in tensor space (A correlation density matrix will be included into each image’s density representation.)

Suppose we have a document and a query, each of them are represented by feature vectors: $d_T = (t_1, t_2, ...t_n)$ and $d_F = (f_1, f_2, ...f_m)$. m and n are visual and textual feature dimensionality respectively. Then the retrieval functions used in our experiment are given as follows.

1. City block (for cbF)

$$sim(d, q) = \sum_{i=1}^{m} |f_i^d - f_i^q| \qquad (32)$$

2. Cosine similarity

   For cosT:

$$sim(d, q) = \sum_{i=1}^{n} t_i^d \cdot t_i^q \qquad (33)$$

For cosT+F and cosT(e)+F:

$$sim(d, q) = \sum_{i=1}^{n} t_i^d \cdot t_i^q + \sum_{i=1}^{m} f_i^d \cdot f_i^q \qquad (34)$$

Our cosine similarity measurement is an approximate cosine similarity, as it can be observed that the similarity score in Equation 34 is not divided by vector length. We report the result of this model rather than the standard cosine similarity for two reasons: the feature values have been normalized within their own feature space; and our experimental results show that the approximate cosine similarity has better performance than the standard one.

3. Measurement in the tensor space.

   Based on quantum measurement, we score a document according to the observable's expectation on the document. With orthogonal assumption of textual basis $|t_i\rangle$ and visual feature basis $|f_i\rangle$, we have:

$$
\begin{aligned}
sim(d,q) &= tr(\sum_i (t_i^d \cdot f_j^d)^2 |t_i f_j\rangle\langle t_i f_j| \\
&\cdot \quad (t_i^q \cdot f_j^q)^2 |t_i f_j\rangle\langle t_i f_j|) && (35)\\
&= trace(\rho_d \cdot \rho_q) && (36)\\
&= \sum_{ij} (t_i^d \cdot f_j^d)^2 (t_i^q \cdot f_j^q)^2 && (37)
\end{aligned}
$$

This shows the same result of transition probability, which is explained as the probability that a system in state d will be found in state q (Aharonov, Albert, and Au 1981), and it is computed as $P(q|d) = |\langle q|d\rangle|^2$. When this classical quantum view is applied to retrieval model, $|\langle q|d\rangle|^2$ can be explained as the probability that a document can be observed containing the information described by the query.

Let us still take the superposed document and query as an example:

$$|d\rangle = \sum_{ij} \gamma_{ij}^d |t_i f_j\rangle,\ |q\rangle = \sum_{ij} \gamma_{ij}^q |t_i f_j\rangle \qquad (38)$$

Then the transition probability between them is:

$$
\begin{aligned}
sim(d,q) &= P(d \rightarrow q) && (39)\\
&= |\langle d|q\rangle|^2 && (40)\\
&= \sum_{i,j} (\gamma_{ij}^d)^2 (\gamma_{ij}^q)^2 && (41)
\end{aligned}
$$

In such case, the measurement on the document density matrix is the same as the inner product of two states, which equals to the cosine similarity of two flattened tensors, where the document and query are represented in a tensor form. This is also our current experimental setting.

Performance Indicators

We use two widely adopted IR performance measures: Average Precision (AP) and Precision at top 10 retrieved documents (P@10). Precision measures the percentage of relevant document in the whole returned document list. However, being able to return the relevant documents in higher rank is also a desirable performance for a retrieval system. Average precision measures both, it is the average of precisions computed at the point of each of the relevant documents in the ranked list:

$$\text{AvgPrecision} = \frac{\sum_{r=1}^{N} (Precision(r) \times \text{relevant}(r))}{\text{number of relevant documents}} \qquad (42)$$

where r is the rank, N is the number retrieved document, relevant(r) a binary function on the relevance of a document on rank r, and Precision(r) is the precision at a given cut-off rank r:

$$\text{Precision(r)} = \frac{|\{\text{relevant retrieved document of rank r or less}\}|}{r} \qquad (43)$$

Note that the denominator in equation 42 is the number of relevant documents in the entire collection, so that the average precision reflects performance over all relevant documents, regardless of a retrieval cut-off.

Currently we focus on testing the effectiveness of the model, and therefore do not include additional efficiency measures. Nonetheless, the tensor model is obviously computationally more expensive than standard approaches. How to reduce the computational cost will be an important issue in our future work.

Experimental Results

Table 1, the list of evaluation result for each run, shows that the pure content based retrieval, especially with simple features, e.g. HSV histogram, has the lowest performance. Pure text retrieval on images are far more better than content based retrieval. However, the content feature can help to improve the text retrieval performance while just concatenated with textual feature without changing retrieval function.

The tensor of visual feature and textual feature can capture certain relationship between the textual and visual features. Even the pure tensor product without taking into account the correlation between text and visual feature, can improve mean AP by 17% compare to cosT+F, and 34 individual queries have better retrieval results. Still for some queries, their APs drop down compared with the cosine similarity on the feature concatenation.

When using text or content feature alone can not retrieve any relevant image, the pure tensor product can not retrieve any relevant image either. This can be observed on queries 06, 24, 30, 41, 49, and 56. This is not a surprise, as when a document does not project to the space spanned by $|t_i\rangle$, it will not project to the space spanned by $|t_i f_j\rangle$ either. Therefore, the pure tensor product will not solve the problem that the images without proper annotation will be ranked low. For example, even if an image has very distinctive house visual features but without the word "house" appearing in its text description, its ranking score will be low with respect to a query whose text contains of "house". However, the same image can be ranked high through the correlation of its visual feature with text "house". This shows the reason why detecting and applying the correlation between visual and textual features is an important aspect of image retrieval, which can bridge the semantic gap of content features.

Unfortunately, our two simple methods for correlating visual and textual features did not bring any improvement to the retrieval results. We observed that some words we associated to the feature dimensions do not match any query

	cos T+F		cos T		cb F		cos T(e)+F		tensor T+F		tensor T(e)+F	
Qid	AP	P@10	AP	P@10	AP	P@10	AP	P@10	AP	P@10	AP	P@10
01	0.1395	0.4000	0.0811	0.4000	0.0070	0.0000	0.1161	0.4000	0.0906	0.6000	0.0890	0.6000
02	0.0038	0.0000	0.0050	0.0000	0.0229	0.2000	0.0094	0.0000	0.0134	0.0000	0.0065	0.0000
03	0.0576	0.0000	0.1646	0.0000	0.0003	0.0000	0.0798	0.2000	0.1993	0.0000	0.1656	0.0000
04	0.0187	0.2000	0.0108	0.0000	0.0022	0.0000	0.0168	0.0000	0.0100	0.0000	0.0144	0.2000
05	0.0065	0.2000	0.0040	0.0000	0.0024	0.0000	0.0160	0.2000	0.0208	0.2000	0.0180	0.2000
06	0.0000	0.0000	0.0000	0.0000	0.0005	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
07	0.4916	1.0000	0.4141	0.8000	0.0025	0.0000	0.4998	0.8000	0.4369	0.8000	0.4338	0.8000
08	0.5173	0.8000	0.5651	1.0000	0.0059	0.2000	0.4934	0.8000	0.3929	0.8000	0.3913	0.6000
09	0.0002	0.0000	0.0002	0.0000	0.0006	0.0000	0.0005	0.0000	0.0000	0.0000	0.0000	0.0000
10	0.2374	0.8000	0.5474	0.6000	0.0012	0.0000	0.2826	0.8000	0.5829	0.0000	0.5816	0.0000
11	0.8284	1.0000	0.0685	0.0000	0.7115	1.0000	0.9135	1.0000	0.7548	0.4000	0.7523	0.4000
12	0.0110	0.0000	0.0185	0.0000	0.0000	0.0000	0.0254	0.2000	0.0449	0.2000	0.0459	0.2000
13	0.1137	1.0000	0.0998	1.0000	0.0001	0.0000	0.0964	0.8000	0.0753	0.8000	0.0758	0.8000
14	0.0032	0.0000	0.0055	0.0000	0.0588	0.2000	0.0236	0.0000	0.0042	0.0000	0.0043	0.0000
15	0.3720	0.6000	0.5572	1.0000	0.0061	0.0000	0.2680	0.0000	0.2313	0.0000	0.2294	0.0000
16	0.1385	0.0000	0.1367	0.0000	0.0004	0.0000	0.1409	0.0000	0.2280	0.6000	0.1926	0.2000
17	0.1611	0.6000	0.1563	0.2000	0.0031	0.0000	0.1571	0.2000	0.2674	0.2000	0.2693	0.2000
18	0.2423	0.6000	0.2378	0.4000	0.0048	0.0000	0.2844	1.0000	0.2962	1.0000	0.2950	1.0000
19	0.0198	0.2000	0.0096	0.2000	0.0139	0.2000	0.0318	0.2000	0.0463	0.2000	0.0459	0.2000
20	0.0055	0.0000	0.0098	0.0000	0.0000	0.0000	0.0096	0.0000	0.0341	0.4000	0.0241	0.2000
21	0.2822	0.2000	0.2669	0.0000	0.0153	0.2000	0.2892	0.0000	0.4504	0.6000	0.3319	0.0000
22	0.0004	0.0000	0.0010	0.0000	0.2804	1.0000	0.0319	0.0000	0.0143	0.2000	0.0114	0.2000
23	0.0212	0.2000	0.0267	0.0000	0.0015	0.0000	0.0891	0.6000	0.1272	0.0000	0.1258	0.0000
24	0.0000	0.0000	0.0000	0.0000	0.0015	0.0000	0.0080	0.0000	0.0000	0.0000	0.0000	0.0000
25	0.0012	0.0000	0.0016	0.0000	0.0019	0.0000	0.0015	0.0000	0.0025	0.0000	0.0025	0.0000
26	0.0000	0.0000	0.0530	0.0000	0.0019	0.0000	0.0000	0.0000	0.0530	0.0000	0.0530	0.0000
27	0.5682	0.8000	0.3932	0.0000	0.1154	0.6000	0.6863	1.0000	0.6657	0.8000	0.6618	0.8000
28	0.1008	0.4000	0.1119	0.6000	0.0087	0.0000	0.0916	0.4000	0.1796	0.8000	0.2029	1.0000
29	0.0975	0.0000	0.1557	0.0000	0.0074	0.0000	0.1139	0.0000	0.1320	0.0000	0.1283	0.0000
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
31	0.0605	0.8000	0.0427	0.2000	0.0018	0.0000	0.0696	0.4000	0.1076	0.8000	0.1071	0.8000
32	0.2017	0.4000	0.2779	0.6000	0.0003	0.0000	0.2204	0.6000	0.2922	0.4000	0.2926	0.4000
33	0.0290	0.0000	0.0001	0.0000	0.0475	0.0000	0.0909	0.0000	0.0001	0.0000	0.0001	0.0000
34	0.0810	0.2000	0.0865	0.2000	0.0019	0.0000	0.0785	0.2000	0.0542	0.2000	0.0537	0.2000
35	0.1765	1.0000	0.2202	1.0000	0.0329	0.2000	0.2133	1.0000	0.2500	1.0000	0.2495	1.0000
36	0.5584	0.8000	0.5392	0.8000	0.0171	0.2000	0.5790	0.8000	0.5766	0.8000	0.5769	0.8000
37	0.1213	0.4000	0.0957	0.2000	0.0490	0.6000	0.1415	0.4000	0.1256	0.4000	0.1090	0.4000
38	0.1698	0.4000	0.1058	0.0000	0.0382	0.0000	0.1976	0.2000	0.2769	0.6000	0.2780	0.6000
39	0.0015	0.0000	0.0007	0.0000	0.0012	0.0000	0.0008	0.0000	0.0008	0.0000	0.0008	0.0000
40	0.0049	0.2000	0.0012	0.0000	0.0027	0.0000	0.0031	0.0000	0.0010	0.0000	0.0010	0.0000
41	0.0003	0.0000	0.0002	0.0000	0.0003	0.0000	0.0006	0.0000	0.0004	0.0000	0.0004	0.0000
42	0.2484	0.0000	0.2798	0.0000	0.0016	0.0000	0.2931	0.0000	0.3256	0.0000	0.3231	0.0000
43	0.2097	0.4000	0.1767	0.4000	0.0231	0.2000	0.2163	0.4000	0.1583	0.4000	0.1479	0.4000
44	0.0429	0.4000	0.0606	0.4000	0.0044	0.2000	0.0643	0.8000	0.0634	0.2000	0.0636	0.2000
45	0.0491	0.4000	0.0243	0.4000	0.0377	0.2000	0.0511	0.4000	0.0698	0.6000	0.0698	0.6000
46	0.0021	0.0000	0.0019	0.0000	0.0075	0.0000	0.0031	0.0000	0.0109	0.0000	0.0066	0.0000
47	0.0143	0.2000	0.0048	0.0000	0.0028	0.0000	0.0286	0.2000	0.0143	0.2000	0.0143	0.2000
48	0.1106	0.0000	0.0931	0.0000	0.0385	0.2000	0.1252	0.0000	0.1805	0.4000	0.1913	0.4000
49	0.0004	0.0000	0.0000	0.0000	0.0172	0.2000	0.0012	0.0000	0.0000	0.0000	0.0000	0.0000
50	0.0919	0.4000	0.0270	0.0000	0.0004	0.0000	0.1550	0.0000	0.1956	0.0000	0.1917	0.0000
51	0.0921	0.6000	0.0867	0.6000	0.1321	0.6000	0.0990	0.6000	0.0896	0.6000	0.0896	0.6000
52	0.0012	0.0000	0.0001	0.0000	0.0003	0.0000	0.0024	0.0000	0.0001	0.0000	0.0001	0.0000
53	0.2254	0.8000	0.1463	0.6000	0.0049	0.0000	0.2656	1.0000	0.2477	0.6000	0.2452	0.6000
54	0.0584	0.2000	0.0575	0.0000	0.0243	0.2000	0.1008	0.2000	0.1033	0.0000	0.0925	0.0000
55	0.1323	0.2000	0.0013	0.0000	0.2829	0.8000	0.2468	0.8000	0.0211	0.4000	0.0212	0.4000
56	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
57	0.9751	1.0000	0.9536	1.0000	0.0004	0.0000	0.9255	0.8000	0.9255	0.8000	0.9073	0.8000
58	0.2491	0.8000	0.2050	0.6000	0.0044	0.0000	0.3272	1.0000	0.3057	0.8000	0.3148	0.8000
59	0.0513	0.0000	0.0727	0.2000	0.0021	0.0000	0.0858	0.0000	0.0918	0.2000	0.0908	0.2000
60	0.2394	0.4000	0.2132	0.4000	0.0691	0.4000	0.2136	0.4000	0.2640	0.4000	0.2389	0.2000
mean	0.1440	0.3167	0.1313	0.2300	0.0354	0.1267	0.1596	0.3133	**0.1685**	0.3067	0.1638	0.2867

Table 1: AP and P@10 for each query

text. As a result, the ranking scores for the images would not change at all. There are some reasons. For example, the annotation of some image does not account the content of the image. e.g., “the destination of the tourist”. The query text sometimes does not associate to any specific the content feature, e.g., “Asian traffic”. Extracting correlation information for each query sample can be a solution. Another reason can be that the simple visual feature such as color histogram is not suitable to be associated with semantic meanings, as (Wang, Hoiem, and Forsyth 2009) also claimed that the association identified by the simple feature normally will not improve the retrieval performance.

Conclusion and Feature Work

In this paper, we introduced a quantum theory inspired multimedia retrieval framework based on the tensor product of feature spaces, where similarity measurement between query and document follows quantum measurement. At the same time, the correlations between dimensions across different feature spaces can also be naturally incorporated in the framework. The tensor based model provides a formal and flexible way to expand the feature spaces, and seamlessly integrate different features, potentially enabling multimodal and cross media search in a principled and unified framework.

Experiment results on a standard multimedia benchmarking collection show that the use of quantum-like measurement on a tensored space leads to remarkable performance improvements in term of average precision over the use of individual feature spaces separately or simple concatenation of them. However, the incorporation of dimension-wise correlation across feature spaces in the tensor model does not lead to performance improvement. Further investigation is needed in this direction.

In current experiments, we assumed each word is orthogonal, but this assumption can be relaxed. We can either replace the synonyms with one representative word, or apply dimensionality reduction in our tensor model. This is also sensible from a practical point of view, as too high dimensionality in textual feature space will make it infeasible to compute ranking scores on a large large collection. Further, we would like to test the tensor product model on a wide selection of visual content features.

References

Aharonov, Y.; Albert, D. Z.; and Au, C. K. 1981. New interpretation of the scalar product in hilbert space. *Phys. Rev. Lett.* 47(15):1029–1031.

Liu, H.; Song, D.; Rüger, S. M.; Hu, R.; and Uren, V. S. 2008. Comparing dissimilarity measures for content-based image retrieval. In *AIRS'08*, 44–50.

Matthew Simpson, Md Mahmudur Rahaman, D. D.-f. S. A. G. R. T. 2009. Text- and content-based approaches to image retrieval for the imageclef2009 medical retrieval track. Working Notes for the CLEF 2009 Workshop.

Min, P. 2004. A comparison of text and shape matching for retrieval of online 3d models - with statistical significance testing. In *In Proc. European Conference on Digital Libraries*, 209–220.

Rahman, M. M.; Bhattacharya, P.; and Desai, B. C. 2009. A unified image retrieval framework on local visual and semantic concept-based feature spaces. *J. Visual Communication and Image Representation* 20(7):450–462.

Tjondronegoro, D.; Zhang, J.; Gu, J.; Nguyen, A.; and Geva, S. 2005. Integrating text retrieval and image retrieval in xml document searching. Advances in XML Information Retrieval and Evaluation.

Wang, G.; Hoiem, D.; and Forsyth, D. 2009. Building text features for object image classifications. In *In CVPR, 2009. 124*.

Yanai, K. 2003. Generic image classification using visual knowledge on the web. In *MULTIMEDIA '03: Proceedings of the eleventh ACM international conference on Multimedia*, 167–176. New York, NY, USA: ACM.