A relational vector space model using an advanced weighting scheme for image retrieval

Jean Martinet\textsuperscript{a,\*}, Yves Chiaramella\textsuperscript{b}, Philippe Mulhem\textsuperscript{b}

\textsuperscript{a}Laboratoire d’Informatique Fondamentale de Lille, France
\textsuperscript{b}Laboratoire d’Informatique de Grenoble, France

\section*{ABSTRACT}

In this paper, we lay out a relational approach for indexing and retrieving photographs from a collection. The increase of digital image acquisition devices, combined with the growth of the World Wide Web, requires the development of information retrieval (IR) models and systems that provide fast access to images searched by users in databases. The aim of our work is to develop an IR model suited to images, integrating rich semantics for representing this visual data and user queries, which can also be applied to large corpora.

Our proposal merges the vector space model of IR – widely tested in textual IR – with the conceptual graph (CG) formalism, based on the use of star graphs (i.e., elementary CGs made up of a single relation connected to some concepts representing image objects). A novel weighting scheme for star graphs, based on image objects size, position, and image heterogeneity is outlined. We show that integrating relations into the vector space model through star graphs increases the system’s precision, and that the results are comparable to those from graph projection systems, and also that they shorten processing time for user queries.

© 2010 Elsevier Ltd. All rights reserved.

\section*{1. Introduction}

Image retrieval has been an active research topic for more than 15 years now (Smeulders, Worring, Santini, Gupta, & Jain, 2000). Historically, a lot of the initial studies have been performed on content-based image retrieval systems: such proposals consider low-level features to be an accurate enough representation of image content for the purposes of retrieval. However, in applications like home photograph management, semantic descriptions of image content are useful (Rodden, 1999; Rodden & Wood, 2003). We have developed an image content description model that represents semantics related to visible objects, as well as to the overall organization of the image.

According to classical models of information retrieval (IR), a document is described by a set of representative index terms. In text retrieval, for instance, index terms are keywords extracted from the collection. Because all index terms in a document do not describe it equally, they are assigned numerical weights. The purpose of a weighting scheme is to give emphasis to important terms, quantifying how well they semantically describe documents and distinguish them from each other.

In order to contain the complex knowledge related to either an image description or a user need, a formalism supporting relations is necessary. In particular, the knowledge representation formalism of conceptual graphs, introduced by Sowa (1984), has been used for image representation by Meckkour (1995) and for image retrieval by Ounis and Pasca (1998).

\* Corresponding author. Tel.: +33 659691191; fax: +33 328778537.
E-mail addresses: Jean.Martinet@lifl.fr (J. Martinet), Yves.Chiaramella@imag.fr (Y. Chiaramella), Philippe.Mulhem@imag.fr (P. Mulhem).

0306-4573/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ipm.2010.10.003

Please cite this article in press as: Martinet, J., et al. A relational vector space model using an advanced weighting scheme for image retrieval. Information Processing and Management (2010), doi:10.1016/j.ipm.2010.10.003
We use conceptual graphs to index the images in the collection, assigning some numerical values to elementary sub-graphs in order to represent the importance of the sub-graphs in the index.

This paper describes an information retrieval model whose image representation is based on star graphs. The remaining sections are organized as follows: in Section 2, we review the state of the art in image indexing and retrieval, from the specific points of view of term weighting and relational descriptions. Section 3 describes how documents are represented in the model presented. Section 4 details the novel weighting scheme for star graphs. Section 5 describes experiments about evaluating the benefits of the weighted star graph approach. Section 6 discusses the impact and limitations of the model. A conclusion and some future works are discussed in Section 7.

2. Related works

Approaches in classical text retrieval are mostly based on using keywords as document descriptors (Salton, 1971). The advantage of dealing with keywords is that the semantics are contained within the keywords themselves, unlike in other media e.g. image or video. For instance, a document in which the word “alpinism” appears is considered as a document about “alpinism”. Consequently this document is considered relevant for a query about “alpinism”.

2.1. Term weighting

Document model keywords in a text IR-system are similar to the words of the user’s natural language. They are automatically extracted, filtered, and weighted to give them more or less importance depending on how well they represent the document content. For instance, in the vector space model (VSM), documents and queries are represented with vectors in a n-dimensional space, where n is the number of terms in the index language. The index of a document \( d_i \) is the vector

\[
\vec{d}_i = (w_{1i}, w_{2i}, \ldots, w_{ni})
\]

where \( w_{ij} \in [0,1] \) represents the weight of the term \( t_i \) in the document \( d_i \). A query is represented by a vector \( \vec{q} = (w_{1q}, w_{2q}, \ldots, w_{nj}) \), where \( w_{jq} \in [0,1] \) is the weight of the term \( t_i \) in the query \( q \). The weight of a term represents its importance in the document and its ability to distinguish between documents in the collection, in order to distinguish relevant documents from irrelevant ones. Most largely used weighting schemes for text retrieval are based on variations of \( tf.idf \) (term frequency × inverted document frequency) (Baeza-Yates & Ribeiro-Neto, 1999; Salton, 1971; Salton & Buckley, 1988; Salton & McGill, 1983). The \( tf \) measures the importance of a term in a document, and is usually defined according to the (normalized) number of occurrences of the term in the document. For instance, the \( tf \) of a term \( t_i \) in a document \( d_j \) is often defined as:

\[
tf_{ij} = \frac{freq_{ij}}{|d_j|}
\]

where \( freq_{ij} \) is the number of occurrences of the term \( t_i \) in the document \( d_j \), and \( |d_j| \) is the number of terms in \( d_j \). The \( idf \) measures the discrimination of the term (Salton & Buckley, 1988; Salton, Wong, & Yang, 1975) (also called resolving power (van Rijsbergen, 1979)), i.e. the ability of the term to distinguish between the documents of the collection. A widely used formula for calculating the \( idf \) of a term \( t_i \) is (Salton, 1971):

\[
idf_i = \log \left( \frac{N}{n_t} \right)
\]

where \( N \) is the total number of documents in the collection and \( n_t \) is the number of documents in which the term \( t_i \) occurs. This weighting scheme is based on two sources. The former represents the local importance of the term in the document and the latter represents the global importance of the term in the whole collection. Weighting schemes are usually based on a combination of these two sources, e.g. \( w_{ij} = tf_{ij} \times idf_i \) in documents, \( w_{jq} = idf_j \) in the queries, and the matching (Retrieval Status Value, RSV) is evaluated with the cosine of their vectors (Baeza-Yates & Ribeiro-Neto, 1999):

\[
RSV(d_j, q) = \cos(\vec{d}_j, \vec{q})
\]

When dealing with image documents, the problem of the semantic gap (Smeulers et al., 2000) arises: because of the distance between the raw signal (i.e. the pixel matrix) and its interpretation, it is difficult to automatically extract an accurate semantic content representation of images. Image retrieval approaches can be divided in two main categories: signal-based approaches and semantic-based approaches.

Signal-based approaches consider low-level image features (e.g. color, texture, etc.) to be image descriptors. Images and queries are represented with feature vectors, the coordinates of which represent the amount of different colors or textures in the image. Hence the numerical values of such vectors have a different meaning than the weights in the VSM, as they are not related to the features’ semantic importance. Another important difference is that the notion of \( idf \) is often not integrated into
these systems. However, a normalization of the dimensions of RGB color space is applied in QBIC (Flickner et al., 1995; Niblack et al., 1993): every coordinate in a vector (or histogram) is divided by the standard deviation of all values in the collection for this coordinate. Hence the mean color vector for a document \( d_j \) is:

\[
\bar{d}_j = \left( \frac{R_j}{\sigma_R}, \frac{G_j}{\sigma_G}, \frac{B_j}{\sigma_B} \right)
\]

where \( R_j, G_j \) and \( B_j \) are respectively R, G and B components of the image \( d_j \), and \( \sigma_R, \sigma_G \) and \( \sigma_B \) are respectively the standard deviations of R, G and B components for the whole collection. This normalization aims to homogenize value distributions for the three dimensions.

Wang, Li, and Wiederhold (2001) and Wang and Du (2001) introduced in SIMPLYcity the region frequency and inverse picture frequency (\( rf, ipf \)), a region-based measure for image retrieval purposes, inspired by the \( tf, idf \) weighting scheme for image retrieval. The images in the collection are segmented into regions, employing several features (e.g. color components). The \( rf \) measures how frequently a region feature occurs in a picture, and the \( ipf \) attaches a higher value to region features that occur in fewer images, which are therefore considered good discriminators.

All regions are assigned an \( rf, ipf \) weight, which is stored for the image matching process. This weighting scheme has been evaluated for an image classification task with 10 categories such as "beach", "mountain" or "building". Results show that a higher classification quality is achieved with this weighting scheme than when using simple color histograms. The basic difference is the integration of the region distribution into the collection, which makes SIMPLYcity closer to semantic-based IR approaches.

The work presented here is inspired by text retrieval, in the sense that we attach a semantic label to (combinations of) objects appearing in images. The objective is to give them more or less importance according to how well they describe the image.

### 2.2. Relational descriptions

Semantic-based approaches integrate a semantic interpretation of the image content. Simple keywords can be used to describe the main elements in an image, e.g. “man” or “sky”. However some information contained in the image cannot be expressed or modeled by keywords themselves (Onnis & Pasca, 1998), such as the semantic and spatial relations between objects, or object attributes. Expression of the spatial relations between objects have been studied in research by Bertino and Catania (1998) and Di Sciascio et al. (2002), in which the authors develop languages to represent the objects’ shape and position in an image. These languages deal with relational image indexing and querying. The semantic interpretation of image content can also be integrated with types of high-level knowledge representation formalism, such as conceptual graphs (CG) (Sowa, 1984) that constitute a formal framework for image representation and retrieval. The CG formalism, like many types of knowledge representation formalisms in artificial intelligence, allows for Boolean truth values only. Numerous studies have been carried out to extend the conceptual graph formalism. Morton (1987) extend Sowa’s conceptual graph theory to include fuzzy concepts, fuzzy referents, and fuzzy operations. Wuwongse and colleagues (1993) extend Morton’s fuzzy conceptual graphs to take into account fuzzy conceptual relations. Maher (1991) proposes a similarity measurement for matching simple fuzzy conceptual graphs, based on fuzzy similarity measurement for concepts. Those studies aim to assign a non-Boolean truth value to a conceptual graph, in order to take account for uncertainty when confronted with imprecise knowledge.

The CG formalism has been used for image representation by Mechkour (1995) and for image retrieval by Onnis and Pasca (1998). In such approaches, the similarity between a document and a query is determined using the graph projection operator that consists of a graph isomorphism, known to be of exponential complexity (Chein & Mugnier, 1992; Mugnier & Chein, 1996). This point impacts the document and query processing time negatively, as pointed out by Mechkour in Mechkour (1995), in which the processing time was close to 2 min for a query, which does not match the usability constraints of an end-user system (Nielsen, 1994; Bouch, Kuchinsky, & Bhatti, 2000). Onnis and Pasca suggest shifting a part of the data processing to the indexing time (i.e. offline), in order to make the query processing algorithm polynomial. However, the indexing time required grows exponentially with the size of the collection. Moreover, this system does not allow users to rank the relevant documents in decreasing order of relevance, but only to organize documents into the three following relevance classes:

1. Total matching
2. Partial matching
3. No matching at all

The approach presented in this paper is an extension of the conceptual graph approach for representing and searching for complex documents, such as images. In particular, this extension consists of using elementary sub-graphs as image descriptors in a vector space representation. Star graphs are made up of a single relation attached to one or more concepts representing the objects in an image. This representation is based on an image model described in the following section.
3. Image model

What do users actually consider significant when describing images, and how can a model reflecting their behavior towards image relevance be developed? According to previous experiments in the context of retrieving non-specific images (Hollick, Smeere, Wielinga, & Worrying, 2004) related to home photographs (Lim, 2001; Mulhem, Lim, Leow, & Kankanahili, 2003), users tend to describe images based on the objects appearing in them, and users seem to find it more intuitive to access images using natural language descriptions, rather than low-level characteristics such as colors or textures (Rodden, 1999; Rodden & Wood, 2003). This explains why our approach focuses on visible objects and not, say, on color regions. It also explains why such visible objects are considered basic entities in our image model. The model in this paper is based on an indexing language composed of star graphs (see Definition 1 below), that are relational descriptors for the images.

3.1. Physical image

Firstly, we introduce the following notions of image and region on which our image model is based. An image is basically considered to be a matrix of pixels.

Definition 1 (Image). An image I is a set of spatially connected pixels organized in a rectangular matrix, corresponding to a two-dimensional projection (or visualization) of a real-world scene. The number of pixels of I is |I|, written n_I. The image represents a set Ω of visible physical objects.

In order to provide a description of the objects visible in an image, we now define the notion of a region.

Definition 2 (Region). A region is defined as a two-dimensional projection (or visualization) of a part of an occluded physical object which is itself part of the scene displayed in an image. Formally, a region r ⊆ I is a non empty set of spatially connected pixels from an image I, r ∈ P(I) where P(I) is the set of non empty spatially connected parts of I. The number of pixels of r is |r| ≥ 1, written n_r.

Definition 3 (Region Segmentation). The set of regions in an image I forms a partition S^I_n ⊂ P(I) of I:

- ∀ r, r_j ∈ S^I_n : r_j ∩ r_j = ∅
- ∪_{r ∈ S^I_n} r = I

The number regions of I is |S^I_n|, written n^{I}_n.

A relation R_I defined on Ω × S^I_n associates physical objects and regions is one-to-n: several regions can be associated to different parts of one (occluded) physical object, and all visible physical objects are associated to at least one region. This relation is shown in Fig. 1.

3.2. Image objects

Our model is based on the notion of image object (IO). This following definition of an image object is inspired from Mechkour (1995):

Definition 4 (Image Object). An image object is defined as a two-dimensional projection (or visualization) of a physical object (or possibly several physical objects) which is (are) part(s) of the scene in an image. Formally, an image object o ⊆ S^I_n is a non empty set of not necessarily spatially connected regions from S^I_n, o ∈ P(S^I_n) where P(S^I_n) is the set of non empty subsets of S^I_n. The number of regions of o is |o| ≥ 1, written n^o, and the number of pixels of o is |o| = ∑_{r ∈ o} n_r, written n^o.

Image objects provide a flexible model for describing image content, as – according to Definition 4 – an IO may correspond to several non-connected regions. Hence, they provide a solution for handling groups of objects in photographs. For instance, they can be used to describe a group of boats in a harbor, a group of trees in a forest, or a group of people in a crowd, with a given level of granularity. This flexibility also makes it possible to take into account the common occurrence in which the object is partially occluded (a car behind a tree, for instance). We will refer further to these two situations as fragmentation cases of an IO.

Definition 5 (Image Object Segmentation). The set of image objects of an image I forms a partition of S^I_n, S^I_0 ⊂ P(S^I_n):

- ∀ o_i, o_j ∈ S^I_0 : o_j ∩ o_j = ∅
- ∪_{o ∈ S^I_0} o = S^I_n

1 Non specific images here means non application-specific images, such as medical or satellite images usually are.
The number image objects of $I$ is $|S^I_0|$, written $n^I_0$. By construction, $S^I_0$ is a partition of $I$. Every region belongs to one and only one image object. Hence, the region partition of the pixels is finer than the image object partition:

$$\forall r \in S^R \exists! o \in S^I_0 | r \in o$$

A relation $R_2$ defined on $S^R \times S^I_0$ associates regions to image objects. This relation is $n$-to-one, as several regions can be grouped into a single image object.

Fig. 2 shows a graphical view of $R_2$. A relation $R_3$ defined on $\Omega \times S^I_0$ connects physical objects to image objects. This relation is also $n$-to-one, since several physical objects can be connected to one image object. Fig. 3 shows a graphical view of $R_3$.

3.3. Interpretation of an image object

An IO is related to one or several semantic interpretations which are defined by labels. Here a label stands for a semantic interpretation of the related IO. In its general definition, a label may belong to any kind of indexing language, as part of a particular indexing model. With regard to images, such labels are usually natural language words ("tree", "house", "sky", "boat", etc.) In some cases, it is not straightforward or relevant to assign a semantic interpretation to certain IOs: blurred, cluttered, or background objects, for example. The indexing language may contain one specific label to be attached to void objects that have no semantic contribution to the index.

Physical objects and IOs are connected through semantic interpretations of IOs. An IO $o$ is connected to a concept\(^2\) $c = [t, \#o]$ whose type $t$ corresponds to the label of $o$ and whose referent $\#o$ is an identifier of the IO. This connexion is one-

\(^2\) A concept as an element of the conceptual graphs formalism.
to-n, since a single object can have several semantic interpretations in the indexing language. The concept type set $T_c$ is organized in a lattice $(T_c, \leq c)$ whose relation $\leq c$ is a generic/specific relation (IsA) between concept types. Fig. 4 shows an example of concept type lattice, in which concept types are organized according to a partial order $\leq c$. For instance, Flower $\leq c$ Foliage.

3.4. Discussion

In linguistics, synonyms are different words or phrases with identical meanings, and polysemy refers to a word or phrase's capacity for multiple meanings. A case in which several IOs have the same interpretation is referred to as visual synonymy and a case in which an IO can have several interpretations is referred to as visual polysemy. While the problem of visual polysemy is raised during the automatic selection of a label (or concept) to be attached to IOs, the model tackles the problem of visual polysemy by allowing several labels to be attached to a given IO.

3.4.1. Automatic label selection: the problem of visual synonymy

The automatic segmentation and labeling of images and video keyframes is prone to error and remains an ongoing problem in computer vision. The challenge is automatically determining a semantic label for a given set of pixels that is prone to great variations. The identification of IOs continues to be a problem, and most of the existing segmentation methods rely on the low-level characteristics of image parts, such as color, intensity, texture, and shape. The last 10 years have witnessed significant improvements in this regard. The methods for segmentation include model-based segmentation (Carson, Thomas, Belongie, Hellerstein, & Malik, 1999), graph partitioning segmentation (Shi & Malik, 2000), clustering methods (Wang et al., 2001; Wang & Du, 2001), and multi-scale segmentation (ter Haar Romeny, 2003). The methods for annotation/recognition include neural networks classifiers (Lim, 2000, 2001), and SVM-based classification (Ayache, QuTnot, & Gensel, 2007; Snoek & Worring, 2005; Vapnik, 1995), for instance using popular SIFT features (Lowe, 2004). In order to avoid the imprecise segmentation process, approaches increasingly rely either on grid-based segmentations in which the images are tessellated in rectangular patches (Bastan & Duygulu, 2006; Lim, 2000, 2001; Martinet & Satoh, 2007; Wang et al., 2001; Wang & Du, 2001), or on interest point detection (Quelhas, Monay, Odobez, Gatica-Perez, & Tuytelaars, 2007; Hoerster, Lienhart, & Slaney, 2007; Quelhas & Odobez, 2006; Zheng, Wang, & Gao, 2006). Local features extracted from patches or around interest points are then quantized and classified according to their visual appearance. In grid-based approaches, contiguous patches sharing the same label can be merged to form larger regions (Lim, 2001). The annotation performance of automatic systems is improving steadily: in Quelhas and Odobez (2006), Quelhas et al. (2007), Vogel and Schiele (2007), the authors achieve annotation precision of about 70% on a 6-class problem. In the case where annotations are automatically derived, information such as the size and the position of corresponding regions can be directly used directly in the model for weighting the indexing terms (see Section 4).

3.4.2. Multi-label justification: visual polysemy

The one-to-n connexion between IOs and concepts originates in visual polysemy, and is justified by the need to represent all possible semantic interpretations of a single, real-world object. While most of the IOs will be connected to a single concept (and therefore a single label) in practice, the model can also represent cases in which it is necessary to define several labels for an IO, such as a window (or a fence, etc.) through which a tree can be seen, or a mirror (or the surface of calm water) reflecting a cloud. In such cases, several concepts are attached to the same IO.

3.4.3. Limitations for other types of relations

On one hand, the advantage of representing concepts in a lattice is that the IsA relation between concepts is integrated into the model (see Definition 8) according to the first order logic interpretation of conceptual graphs subsumption, because the generic/specific relation of concepts corresponds to a hyponym–hypernym relation of subsumption for the labels.

Please cite this article in press as: Martinet, J., et al. A relational vector space model using an advanced weighting scheme for image retrieval. Information Processing and Management (2010), doi:10.1016/j.ipm.2010.10.003
On the other hand, one limitation of the model is that it represents neither the synonym/antonym relations nor holonym/meronym relations (partOf), since the concept lattice does not represent these relations. Indeed, the concept lattice is part of the conceptual graph formalism, and represents only hyponym–hypernym relations between labels. Our model conforms to these relations in order to be consistent with the CG formalism and the graph projection. As a consequence, the issue of selecting the best label to be attached to an IO among all concepts and their possible relations (e.g., should a wing be labeled with “wing” or “airplane” – or should a forest be labeled with “forest” or “tree” in a fragmented case) is an unresolved problem in IR that lies outside of the scope of this paper. However, the model makes it possible to use both labels, which is consistent with the fact that a given image displaying wing is likely to be relevant to a query about both “wing” and “airplane”.

Nevertheless, the granularity and the richness of the indexing language should match the image database domain, the labeling strategy being dedicated to fulfilling retrieval needs.

3.5. Relational vector space of star graphs

In order to allow relational indexing, concepts can be put into a relation with conceptual relations (Sowa, 1984). Like the set of concepts, the set of conceptual relations $T_r$ is organized in a lattice $(T_r, \subseteq_r)$. Fig. 5 shows an example of a relation lattice in which relations are organized according to a partial order $\subseteq_r$. See for instance: standing $\subseteq_r$ position.

Concepts are connected to a relation by the means of star graphs to denote the relationship between them.

Definition 6 (Star graph). A star graph is an elementary conceptual graph consisting of one and only one relation $r$, connected to $a_i$ concepts $c_i (a_r \geq 1$ is the arity of $r$), with linear writing: $(\tau(c_1, c_2, \ldots, c_n))$.

A star graph (SG) can be either specific (when composed of specific concepts$^3$) or generic (when composed of generic concepts). The indexing vocabulary is defined according to the canonical basis, containing the signatures of the relations from $T_r$.

Definition 7 (Indexing vocabulary). The indexing vocabulary is composed of all generic star graphs derived from the canonical basis star graphs by restricting the relation, or one or several concepts.

In other words, the indexing vocabulary contains all possible restrictions of the star graphs from the canonical basis.

Example of star graphs are given in Fig. 6. The following star graphs:

- (rowing(\textit{Jean}:#2))
- (on(\textit{Boat}:#4, \textit{Sea}:#3))
- (between(\textit{Boat}:#4, \textit{Matthieu}:#1, \textit{Jean}:#2))

are specific as they contain specific concepts.$^4$ A star graph is said to be generic when it contains generic concepts only. Our definition of a star graph is slightly different from the one proposed by Chein and Mugnier (1992) and Mugnier and Chein (1996), who define a star graph as a relation attached to concept types. In our definition, relations are attached to concepts (that is to say a specific star graph). However, in essence, our definition of a star graph and the definition by Chein and Mugnier share the same underlying support:

- the generic star graphs are seen as building blocks for generating conceptual graphs by making them specific, and then by performing joint operations on them to build the graph,
- the specific star graphs in the index of a document are seen as instances of generic star graphs that are the index terms of the presented model.

---

$^3$ A specific concept is a concept whose referent is an individual, and not the universal referent ($\ast$).

$^4$ A specific concept has an individual referent, unlike a generic concept whose referent is the universal referent ($\ast$).
The star graphs given in Fig. 6 are taken from the description of the left image in Fig. 7. While the concepts of these star graphs could label the image on the right, the star graphs do not apply. Star graphs are more expressive than simple keywords, and less complex by construction than large conceptual graphs. Hence we consider them as a satisfying trade-off between the lack of expressivity of keywords and the processing complexity of conceptual graphs.

The set of star graphs (SGs) extracted from the collection of conceptual graphs has a lattice structure, composed with sub-lattices of relations sharing the same arity (number of concepts). Its partial order relation $\preceq_{SG}$ is the graph projection (Sowa, 1984). This lattice is limited at the top by $\top_{SG}$ and at the bottom by $\bot_{SG}$. It has the general outline shown in Fig. 8. In this figure, there are three sub-lattices, containing star graphs respectively with two, three, and four concepts.

Documents and queries are represented in our model by vectors of generic star graphs. For instance, if a document is described by the specific star graphs of Fig. 6, then its index will include the following generic star graphs: $\langle\text{rowing}\langle\text{Jean}\rangle\rangle$, $\langle\text{on}\langle\text{Boat, Sea}\rangle\rangle$ and $\langle\text{between}\langle\text{Boat, Matthieu, Jean}\rangle\rangle$.

The referents are not included in the indexing language as they refer to particular instances of concepts.

---

**Fig. 6.** Star graphs of unary, binary and ternary relations (from left to right).

**Fig. 7.** Example of two images likely to be labeled with “Jean”, “Matthieu”, “Boat”, and “Sky”. While these four labels do not make it possible to distinguish between these images, the star graph helps differentiating them since they describe only the image on the left.

**Fig. 8.** General outline of the star graph lattice.
3.6. Image index

Automatic document expansion is an important part of the indexing process in our model. This document expansion adds some related index terms to the index of the original document. When a document is indexed by a given index term, it is also implicitly described by all index terms that are generic to the given index term. For instance, a document that is originally described by the term \(\text{left}(\text{Boat}, \text{Pine-tree})\) is also implicitly described by the term \(\text{left}(\text{Boat}, \text{Tree})\). If, for example, the concept type lattice contains the information \(\text{Pine-tree} \preceq \text{Tree}\). The automatic document expansion process adds the second term above (if not already included) to the index of the document. The addition of a term to the index of a document means changing its weight to a non-zero positive value.

**Definition 8** (Automatic document expansion). Let \(\mathcal{SG}\) be the set of specific star graphs indexing a document. The expansion of this document is defined by the set of specific star graphs \(\mathcal{SG}':\)

\[
\mathcal{SG}' = \mathcal{SG} \cup \{ \mathcal{sg}_j \mid \mathcal{sg}_i \in \mathcal{SG} : (\mathcal{sg}_i \preceq \mathcal{sg}_j) \}
\]

The set \(\mathcal{SG}'\) is the index of the document.

The operation of adding an extra term to the index means changing its weight from 0 to a non-zero value. If the term that is added already belonged to the index, its weight will still be increased. Fig. 9 shows the index of a document before (above) and after (below) the automatic document expansion. Dimensions corresponding to generic dimensions of original dimensions are affected non-zero values, namely the weight of the term \(\text{row}(\text{Person})\) is 0 before the document expansion, and becomes \(w_{k,1} > 0\) after the document expansion, assuming that Jean, Matthieu and Caroline are specific concept types of Person.

3.6.1. Justification

The justification behind the operation of document expansion as part of the model originates in the first order logic interpretation of conceptual graphs subsumption. Indeed, a projection of a graph \(g\) into a graph \(h\) exists if and only if \(h \Rightarrow g\) (Chein & Mugnier, 1992; Sowa, 1984). Since our model uses a lattice to represent the indexing vocabulary, the set of star graphs is not flat (i.e. a bag of terms). It is necessary to guarantee the consistency of the vector match with regard to the logical implication. Hence, this pre-processing step ensures a non null term intersection between indexes containing star graphs that are comparable to the relation \(\preceq_{\mathcal{SG}}\).

3.6.2. Limitation

The implementation of automatic document expansion in the model is results exceptions problems, inherited from the standard automatic document expansion in text retrieval. For instance, if the concept lattice states \(\text{Penguin} \preceq \text{FlyingAnimal}\),

![Diagram](image_url)

**Fig. 9.** Vector of the document before (above) and after (below) the document expansion.

then the automatic document expansion would generate index terms with “FlyingAnimal” for images containing a “Penguin”.

To avoid this still unresolved problem, a disambiguation step is necessary.

One way of avoiding the problem is to take special care in the design of the ontology that results in the concept lattice. Another solution is to use Latent Semantic Indexing (LSI) techniques (Deerwester, Dumais, Furnas, Landauer, & Harshman, 1990; Hull, 1994). LSI is an IR method based on a singular value decomposition for identifying patterns in the relationships between terms and concepts in an unstructured collection of text. LSI is based on the principle that words that are used in the same context tend to have similar meanings. A key feature of LSI is its ability to extract the conceptual content of a body of text by establishing connections between terms that occur in similar contexts. Applying LSI to concepts and/or star graphs could potentially resolve this issue.

4. Term weighting

We have developed the image model, in which documents and queries are represented with star graph vectors. In the context of IR, assigning weights to index terms in order to emphasize important terms is a central problem. The purpose is to quantify how well they describe documents semantically and distinguish between them. Previous section defined the indexing vocabulary; this section discusses the aboutness of an image considering a given index term, i.e. the importance of the index term regarding the image. The challenge is assigning a weight to an index term for the purpose of measuring the extent to which the image containing the corresponding star graph is relevant for a query about the index term.

In order to define such a weight of an index term (i.e. a star graph), it is necessary to define first the notion of importance of concepts. This importance is directly related to the importance of their associated image objects. In our approach to image indexing, such image components are intended to provide a basis for semantic image indexing, but also potentially to feature-based indexing (or signal indexing) of images.

These definitions are related to the notion of image content in a similar way to the standard notion of term importance in the context of text retrieval. The notion of occurrence – now very classical when considering the weighting of index terms for textual documents – is much less intuitive in the case of images. Images are 2D data and in this context it should be understood that the relevance of an image showing boats is not only related to the fact that it represents one or more distinct boats. In that regard, other 2D perceptive factors are possibly more important, such as the spatial position or the size of these boats.

4.1. Image objects importance

Considering an image \( I \) and an image object \( o \) of \( I \), \( o \) is labeled by a concept type \( t \) (see Section 3.3). The importance of \( o \) regarding \( I \) is directly related to the aboutness of \( I \) considering \( t \). For example, one will consider that an IO representing a boat (hence \( t = “boat” \)) in a given image is important if most users would consider that this image is about a “boat”. In our opinion, aboutness is a notion mainly local to a given document. It should not be confused with relevance, a notion that usually compares a particular document to the whole corpus. The three criteria outlined here are local to images and thus cannot be directly assimilated to relevance evaluation. A relation between these two notions is somewhat illustrated by the \( tf.idf \) weighting model: given a term \( t \) of document \( D \), \( tf(t) \) stands for the aboutness of \( D \) considering \( t \), while \( tf.idf \) is an estimate of the relevance of \( D \) considering \( t \).

In earlier studies (Martinet, Chiaramella, & Mulhem, 2005, 2008), we propose and validate a definition of the importance of media objects, and more specifically image objects. We give a general overview of this work and refer interested readers to the original publications for more details.

Connected to an image region (or a set of image regions), an IO is basically a 2D set of pixels whose basic geometrical characteristics (area, position, etc.) can be easily calculated. For instance, the area can be estimated by summing up the surface of polygons delimiting the IO, and the position can be assimilated to the center of the bounding box of the IO.

According to the weighting model defined in Martinet et al. (2005, 2008), the importance of an IO is determined by the following three criteria:

- **Criterion 1 (Size):** The importance of an image object varies in the same way as its size \( S \).
- **Criterion 2 (Position):** The importance of an image object is maximal when its position \( P \) is at the center of the image, and decreases when its distance from the image center increases.
- **Criterion 3 (Homogeneity):** The importance of an image object varies in the same way as the homogeneity \( H \) of its embedding image.

The modeling of the criteria is based on probability theory and Shannon information theory (Shannon, 1948), which represent a formal framework suited for IO importance modeling.

The importance of an IO \( o \) in an image \( I \) has been defined as:

\[
\text{importance}_o(o) = S_{norm}(o,I) + P_{norm}(o,I) + H_{norm}(I) 
\]

(4)

where \( S_{norm}, P_{norm} \) and \( H_{norm} \) are the normalized values of criteria \( S, P \) and \( H \) respectively. The normalization is done with:

Please cite this article in press as: Martinet, J., et al. A relational vector space model using an advanced weighting scheme for image retrieval. Information Processing and Management (2010), doi:10.1016/j.ipm.2010.10.003
where $\min_value$ (resp. $\max_value$) is the minimum value (resp. the maximum value) of the criterion value found in the whole collection. Since the three criteria might have different variation ranges on different collections, the normalization is applied in order to align both lower and upper bounds (Lee, 1997).

The criteria $S$, $P$, and $H$ are modeled in the following way:

1. Human perception of surfaces being logarithmic rather than linear (Rojet & Schwartz, 1990), we define a normalized surface according to the following formula:

   \[
   S(o, I) = \frac{\log(n_o^p)}{\log(n_i^p)} = \log_{n_i^p}(n_o^p)
   \]

   where $n_o^p$ is the number of pixels in $o$, $n_i^p$ is the number of pixels in $I$, and $\log_{n_i^p}$ is the logarithm in base $n_i^p$. $S(o, I)$ is an increasing function of $n_o^p$ (according to the first criterion) that has a logarithmic variation with values in $[0, \log_{n_i^p}(n_i^p)]$, that is $[0,1]$. 

2. The position criterion $P$ is integrated into our model through a non-uniform probability density function:

   \[
   P(o, I) = p_i(o)
   \]

   where $p_i$ is a probability distribution that gives higher probability values to IOs in the center of an image $I$. Some examples of such a distribution are given in Fig. 10, where $I$ is a simplified image seen as a 1D segment, and a 1D distribution is shown above $I$. Two image objects $o_1$ and $o_2$ are represented as parts of the segment, and their associated probabilities correspond to the black areas below the probability density curves. These distributions would correspond – for 2D images – to (a) a flat distribution, (b) two plateaux: a square/circular high level in the center and a low level for the remainder if the image, (c) a conic distribution, and (d) a bell-shaped (Gaussian-like) distribution.

   The three distributions (b)-(d) meet the requirements related to the second criterion, as the probability associated with $o_1$ is higher than the one associated with $o_2$. The position values belong to $[0,1]$ and they are greater for IOs in the center of the image according to this criterion.

3. With regard to the homogeneity criterion, let us consider normalized sizes of IOs as probabilities, as they verify all classical Kolmogorov properties of a probabilistic distribution and are therefore isomorphic to a probability. The probability associated with an IO $o$ corresponds to the probability that a pixel taken randomly from the image belongs to $o$:

   \[
   p(o, I) = \frac{n_o^p}{n_i^p}
   \]

   Based on this probability, we can define a spatial entropy $SH$ (according to Shannon (1948)) calculated from the spatial distribution of IOs in an image, in order to represent the “disorder” in the image:

   \[
   SH(I) = \sum_{o \subseteq I} p(o, I) \times -\log(p(o, I))
   \]

   The term $-\log(p(o, I))$ represents the amount of information held by $o$ in the context of $I$. IOs have a minimal size, $p(o, I) \in [\frac{1}{n_i^p}, 1]$ and consequently: $-\log(p(o, I)) \in [0, \log(n_i^p)]$. Entropy values are larger when there are many objects with the same size in an image and when one object in particular is less visible. Entropy values are smaller when there are fewer objects with different sizes, when one object is bigger than the others and therefore more easily visible in the image. To be consistent with our last criterion, the homogeneity value is defined as the complement to 1 of $SH$ to which we apply a normalization factor. The homogeneity $H$ is defined according to the following formula:

   \[
   H(I) = 1 - \frac{SH(I)}{SH_{\text{max}}(I)}
   \]

   where $SH_{\text{max}}(I)$ is the maximum value of the spatial entropy, corresponding to the virtual case where the image is composed of $n_i^p$, 1-pixel IOs. The homogeneity values are large for homogeneous images according to the last criterion, and they belong to $[0,1]$. This value is the same for all IOs in one image.

Fig. 10. Uniform probability distribution (a), and examples of non-uniform probability distributions (b-d).
Hence, our definition of IO importance combines the three criteria that have been experimentally validated in Martinet et al. (2005, 2008) to reflect the aboutness of images with regard to the semantic interpretation of IOs. This definition of IO importance provides the foundations of the weighting model for generic star graphs that are the index terms of the presented model.

4.2. Weighting scheme for index terms

Based on the definition of IO importance, we define the weighting model for the indexing vocabulary. The model is an extended and refined definition of the model presented in Martinet, Chiaramella, Mulhem, and Oonis (2003, 2003). The main differences are the criteria by which the importance of individual concepts is defined and their combination.

4.2.1. Local importance of a star graph in a document

Specific star graphs connected to an image are composed of specific concepts whose referents are the image objects. Hence the importance of a specific star graph is related to the importance of its IOs concepts. Let us first associate the importance of a concept $c_k = [t, \#o]$ with the importance of its IO $o$:

$$\text{importance}_{ij}(c_k) = \text{importance}_{ij}(o)$$

A function combining the importance values of the concepts must satisfy the following requirements:

- The importance value of a star graph should increase when the importance values of its concepts increase.
- The arity of the relation of a specific star graph should not influence the importance value.

These requirements are satisfied by the mean value. We can define the local importance of a specific star graph $s$ of an image $j$ as:

$$\text{local-specific}_{ij} = \frac{1}{\text{arity}_i} \sum_{k,s \in s} \text{importance}_{ij}(c_k)$$

The local importance of an index term $i$ (generic star graph) for an image $j$ is a function of the importance values of specific star graphs associated with the index term. Inspired by text IR famous weighting schemes for the $tf$ value of a keyword, we define the local importance of an index term as the normalized number of occurrences of its corresponding specific star graphs:

$$\text{local}_{ij} = \frac{\sum_{s \in s} \text{local-specific}_{ij}}{\max_k \text{local-specific}_{ij}}$$

where $s$ refers to the specific star graphs associated with $i$ (written $s \leftrightarrow i$) and $k$ refers to all specific star graphs of the image $j$.

4.2.2. Global importance of a star graph in a collection

In traditional approaches to text vector space models, the global importance of an index term aims to emphasize the impact of the most discriminating index terms (e.g. the use of an $idf$ value). This factor for a star graph $i$ is calculated using the classical formula:

$$\text{global}_{i} = \log \left( \frac{N}{n_i} \right)$$

where $N$ is the number of documents in the collection, and $n_i$ is the number of images of the collection that are described by the star graph $i$.

An important aspect of this factor is that it is applied after the document expansion (see Definition 8). Hence, the semantics are slightly different than the traditional $idf$ (Martinet et al., 2003, 2003), as the index contains all generalizations of original star graphs defined by the indexer.

The traditional $idf$ of a keyword $i$ is calculated by counting the number of documents in which the keyword $i$ occurs, while our global, is calculated for the generic star graph $i$ by counting not only the documents in which $i$ occurs, but also the documents in which any star graph generic to $i$ occurs.

For instance, in order to calculate the global importance value of the term $\langle \text{left(Boat,People)} \rangle$, not only is it necessary to take into account the documents indexed by $\langle \text{left(Boat,People)} \rangle$, but it is also necessary to consider the documents indexed by $\langle \text{left(Sailboat,People)} \rangle$, $\langle \text{left(Boat,Jeans)} \rangle$, or $\langle \text{left(Boat,Matthieu)} \rangle$ (assuming that the concept type lattice contains the following relations: $\text{Sailboat} \subseteq \text{Boat}$, $\text{Jeans} \subseteq \text{People}$, and $\text{Matthieu} \subseteq \text{People}$). Indeed, all of these documents are implicitly indexed by $\langle \text{left(Boat,People)} \rangle$, which is made explicit by the automatic document expansion process (see Definition 8).

4.2.3. Weight of a star graph

The above definitions of the local importance and the global importance of a star graph allow us to define a weighting scheme for star graphs that consists of combination of both values.
According to the classical weighting approaches of the VSM for textual documents (see Section 2), we define the weight of a star graph \( i \) in a document \( j \) based on the following sources: the local, value that is calculated for the star graph \( i \) in the document \( j \), and the global, value that is calculated for the star graph \( i \), independently of the document in which it appears:

\[
w_{ij} = \text{local}_{ij} \times \text{global}_i
\]

Hence, the weight of a star graph combines its local importance in the document and its global importance in the collection.

In the context of our relational vector space model, the users’ queries, like the documents, are represented with star graphs vectors. The weighting scheme for query star graphs is based on their global importance in the collection:

\[
w_{iq} = \text{global}_i
\]

Fig. 11 shows an example of two images from the collection Coll-1 (see the description in the next section) with local and global values for the index term (Horse, Tree). The details of \( S, P, \) and \( H \) values are given for the concept “Horse”. The real surface (not using the log) is also given for information between braces. Note that the \( H \) value is defined for the image itself, and therefore is the same for all concepts. Also note that the global value is defined for a given term in the whole collection. This term is given a high global value (5.296) because only 4 out of the 798 images in the collection are indexed with this term. The global values for terms in this collection range from 0.167 for the term (touch (Natural_Object, Natural_Object)) that belongs (after document expansion) to 675 images, to 6.682 for the term (touch (Car, Flower)) that belongs to only one document. While the first term is considered not useful for discriminating documents in the collection, the second term is very useful to select relevant documents in the collection.

5. Experiments

In previous sections, we develop an image model for representing documents based on vectors of generic star graphs, which are considered the indexing vocabulary. A weighting scheme for star graphs has also been introduced as a part of the retrieval model. This experiment section is aimed at:

- measuring the improvements in performance yielded by the operation of expansion described in Section 3.6,
- evaluating the benefits of using star graphs as compared to simple concepts in the indexing vocabulary,
- comparing the difference in precision between a star graph vector space system and a conceptual graph total projection system.

Fig. 11. Example of local and global values for two images.
5.1. Image test collections

For the purpose of evaluating the presented model, we have used two image collections. The first one, which we will refer to as “Coll-1”, is composed of about 800 personal photographs from the authors. The images have been segmented and indexed manually by defining polygons corresponding to real-world objects, and assigning a semantic label to them. This collection contains a wide range of the authors' holidays photographs. A set of 24 queries has been designed for this collection, containing an equal number of queries with one keyword (or one concept, e.g. “a boat”), several keywords, one relation (or one star graph, e.g. “a water stream under a leafed tree”), and several relations. Fig. 12 shows a sample of images from Coll-1.

The second collection (“Coll-2”), contains about 2300 images that have been indexed automatically (Lim, 2001). The automatic indexing process is based on an artificial neural network, that is trained to classify 20 × 20 image blocks into categories such as rock, building, and water. Regions (or blobs) are formed by aggregating spatially connected blocks sharing the same label. For comparison purposes, we have used the same query set as in the experiments presented in Mulhem and Lim (2002), in which authors have used a set of 24 queries. Fig. 13 shows a sample of images from Coll-2.

For both collections, the importance values of image objects are computed according to Eq. (4). Spatial and position relations have been automatically extracted by considering the barycenter of the regions, and the resulting star graphs are weighted according to Eq. (7) (and Eq. (8) for query star graphs). The relations in the considered lattice include:

- an indication of the absolute location of IOs in the image (such as one IO “is located in such part of the image”, the images being tessellated with a 5 × 5 grid),
- the information of whether an IO is close to (and touches) the border of the image (such as one IO “touches the right border of the image”), and
- an indication of the relative position of IOs (such as one IO “is located below” another IO).

Fig. 12. Sample of images from Coll-1.
These relations are 2D relations, that are automatically computed based on the location and surface of the bounding boxes of IOs. The automated detection of 3D relations between real-world objects is more problematic since IOs are a 2D projection of these objects onto the image. A finer interpretation is necessary to integrate 3D relations. This step would require a 3D scene analysis to extract depth information, by using a probabilistic modeling of the background/foreground over a sequence of images (Li, Huang, Gu, & Tian, 2004), analyzing focus maps (Czuni & Csordas, 2003; Malik & Choi, 2008), using occlusion information (Merchán, Vázquez, Adán, & Salamanca, 2008), or even use a stereoscopic camera (François & Chupeau, 1997).

The Table 1 shows some statistics for both collections. These two collections have been selected because they differ in the content, in the size (Coll-2 is three times as big as Coll-1), in the indexing process (manual and automatic), and in the size of the vocabulary (the vocabulary for Coll-1 is about 10 times as big as the one for Coll-2).

These two collections have been selected because they differ in the content, in the size (Coll-2 is three times as big as Coll-1), in the indexing process (manual and automatic), and in the size of the vocabulary (the vocabulary for Coll-1 is about 10 times as big as the one for Coll-2).

Table 1
Statistics for the two collections.

<table>
<thead>
<tr>
<th></th>
<th>Coll-1</th>
<th>Coll-2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of documents</td>
<td>798</td>
<td>2382</td>
</tr>
<tr>
<td>Number of queries</td>
<td>24</td>
<td>24</td>
</tr>
<tr>
<td>Number of concepts in vocabulary: original</td>
<td>237</td>
<td>26</td>
</tr>
<tr>
<td>Number of concepts in vocabulary: extended</td>
<td>253</td>
<td>38</td>
</tr>
<tr>
<td>Number of star graphs in vocabulary: original</td>
<td>15,868</td>
<td>2000</td>
</tr>
<tr>
<td>Number of star graphs in vocabulary: extended</td>
<td>53,007</td>
<td>7361</td>
</tr>
<tr>
<td>Number of concepts per document: original</td>
<td>6.94</td>
<td>3.01</td>
</tr>
<tr>
<td>Number of concepts per document: extended</td>
<td>16.16</td>
<td>8.81</td>
</tr>
<tr>
<td>Number of star graphs per document: original</td>
<td>75.14</td>
<td>18.74</td>
</tr>
<tr>
<td>Number of star graphs per document: extended</td>
<td>907.58</td>
<td>317.74</td>
</tr>
<tr>
<td>Number of concepts per query: original</td>
<td>1.67</td>
<td>1.58</td>
</tr>
<tr>
<td>Number of concepts per query: extended</td>
<td>31.62</td>
<td>7.29</td>
</tr>
<tr>
<td>Number of star graph per query: original</td>
<td>2.37</td>
<td>2.5</td>
</tr>
<tr>
<td>Number of star graph per query: extended</td>
<td>267.58</td>
<td>38.58</td>
</tr>
</tbody>
</table>

![Fig. 13. Sample of images from Coll-2.](image-url)
5.2. Evaluation protocol

We have implemented different systems, and have compared the retrieval performances in terms of Mean Average Precision (MAP) values and Recall–Precision (R–P) curves. We have used trec_eval tool provided by NIST to produce the results.

The first system (VSM-C) is a vector space system based on concepts only, which corresponds to an implementation of the traditional VSM for text. Star graphs are not used in this system. The second system (VSM-SG) is a vector space system based on star graphs, as defined in Section 3. Both VSM-C and VSM-SG can be run in original space or in extended space, where the operation of expansion adds some new dimensions to the vector space, and new terms to the indexes. In addition, the weighting scheme defined in Eqs. (7) and (8) are used to weight star graphs.

The third system, called DIESKAU (Mulhem & Lim, 2002), is a CG-based system which uses the total graph projection operation to select relevant images for a query. A detailed analysis of the evaluation results is given below.

5.3. Impact of the vector expansion

Figs. 14 and 15 show the MAP values for concepts and star graphs according the different weighting schemes in both original and extended spaces, for Coll-1 and Coll-2 respectively. We can see that for both collections, the MAP value in the extended space is about twice as high as the one in the original space.

For both Coll-1 and Coll-2, the retrieval performance has increased by a ratio of 2 when using the extended documents and queries for the concepts, and by a ratio of 3 for star graphs.

5.4. Relations increase precision in extended space

We compared the results of VSM-C and VSM-SG to measure the increase in precision brought about by integrating relations into the VSM. The results given in Fig. 16 show the MAP value difference between the concept-based system and the
star graph-based system for Coll-1 and Coll-2 in extended space. These results indicate that for Coll-1 (Fig. 16 above) relations significantly increase the precision of the VSM in extended space, from 0.461 to 0.585, or a 27% relative gain. This can also be seen in the R–P curves given in Fig. 19 above by comparing ‘BEST EXTENDED STAR GRAPH’ and ‘BEST EXTENDED CONCEPTS’. In this bar chart, we also display the MAP for DIESKAU of 0.497.

For Coll-2 (Fig. 16 below), the results show that there is almost no difference – see the scale of the chart – in terms of MAP between VSM-C (0.234) and VSM-SG (0.227). The R–P curves given in Fig. 19 (compare ‘BEST EXTENDED STAR GRAPH’ and ‘BEST EXTENDED CONCEPTS’ in the lower chart) confirm that there is no significant gain with VSM-SG. This result is due to the two following reasons:

- the reasonably small size of graphs in Coll-2 documents and queries (on average 3.01 concepts/document and 1.58 concepts/query),
- the large proportion of non-relational query in this collection (12 out of 24).
However, if we select from among all queries the subset of 12 relational queries, the increase in precision becomes significant (see Fig. 17): 9% from 0.254 (VSM-C) to 0.278 (VSM-SG). With this result, we also notice that the MAP for VSM-SG remains comparable to that of VSM-C when using all queries, meaning that the star graph representation is still valid for non-relational queries.

For Coll-2, the MAP values for DIESKAU show a slight increase in precision when using the subset of relational queries (0.136), compared to when all queries are used (0.129).

5.5. Star graphs versus conceptual graphs

In this section, we discuss the differences in search precision between the star graph approach (VSM-CG) and the conceptual graph approach (DIESKAU) more precisely. Figs. 18 and 19 show the R–P curves of these systems for Coll-1 and for Coll-2, respectively.

The R–P curve for Coll-1 shows that for small recall values, the graph projection system (DIESKAU) has a higher precision than the star graph-based system. Indeed, this system is precision-oriented since it uses whole conceptual graphs as document indexes. Also, because the total graph projection is used to select relevant images, the system finds fully matching documents accurately. However, the star graph system does not keep the joint information between indexing star graphs. For instance, the joint information on \([\text{Tree} : \#1] \rightarrow (\text{left}) \rightarrow [\text{Building} : \#2]\)

and

\([\text{Tree} : \#1] \rightarrow (\text{above}) \rightarrow [\text{Car} : \#3]\)

is lost in the vector space representation. Indeed, these two star graphs are part of the index, which could describe both ‘a tree is located to the left of a building and above a car’ (the original description contained in the whole graph with the joint information) and ‘a tree is located to the left of a building, another tree is located above a car’. Hence, documents containing both star graphs are retrieved by the star graph system regardless of this joint information, which potentially judges relevant documents and partially relevant documents to be fully relevant.

For recall values over 0.5, DIESKAU shows a drop in precision, and the precision of the star graph-based system remains reasonably high. This drop is a consequence of the rigidity of the projection system: while both the concept-based system and the star graph-based system retrieve partially relevant documents, the total projection-based system misses these
documents. From this point of view, the vector space system is more flexible than the graph projection system. This result confirms that the graph projection system is precision oriented while the vector space system is recall oriented. This lack of flexibility is more explicit for Coll-2, where the R–P curves show that DIESKAU’s results are lower than the star graph system.

Besides, the R–P curves given in Figs. 18 and 19 confirm the following results explained in Sections 5.4 and 5.5:

- the large increase in precision brought by the operation of expansion for both collections and
- the larger difference in precision between VSM-C and VSM-SG for Coll-1 than for Coll-2.

6. Discussion

The results in the previous section show that the operation of document expansion yields considerable gains in precision, with a ratio of about 2 for concepts and about 2.5 for star graphs. Moreover, integrating relations in the document index and query representation is an important aspect of image retrieval. It helps to increase search precision for relational queries, while not affecting non-relational queries. Last but not least, the overall retrieval performance of our system is of equivalent quality to that of a graph projection system, while query processing time is much shorter (linear) in our system.

In this paper, we have not addressed the validation of the Local weighting model (see Eq. (5)), since this specific part of the model been assessed in Martinet et al. (2005, 2008).

6.1. Impact of local value

In the experiments in this paper, the small differences that can be noticed in the MAP values are the result of minor re-rankings of documents since all different weighting settings of a given system result in the same set of retrieved images. The reason for this is that the standard recall–precision measures evaluate systems based on a ground truth about the binary relevance of documents. Consequently internal re-rankings within the set of retrieved relevant documents have no (or very little) effect on the final MAP value.

For instance, if the result of one system setting – say Boolean – is:

\[ \text{rank}_{\text{Boolean}} = [4, 3, 1, 2, 5] \]

(that is to say the sorted list of images), and the result of the same system with another setting – say Local – is:

\[ \text{rank}_{\text{Local}} = [4, 1, 2, 3, 5] \]

then (assuming that the only relevant documents are 1–3) the MAP value is identical for both settings. This can be seen in Figs. 14–16: the MAP values are very similar for all weighting schemes, although there are small, non-significant variations.

However, the quality of the rankings produced with the Local weighting model is closer to users’ perception of relevance, since this model is based on Eq. (4), which has proved to be a good estimator of this perception.

Besides, it should be noticed that the sparseness of the vector spaces – in comparison to the size of the collections – makes it difficult to evaluate the weighting method directly. Indeed, in Coll-1 (resp. Coll-2), the star graph vector space is 53,000 (resp. 7000) in dimension, while the collection contains about 800 (resp. 2000) images. In Coll-1 in particular, this difference...
of two orders of magnitude makes the occurrence or absence of a star graph in a document index decisive for the retrieval, regardless to its weight.

6.2. Impact of global value

We have found that the Global part of the weighting scheme has no significant impact on the retrieval results. In the original space, the best weighting scheme seems to be Local x Global, while in the extended space slightly better results are achieved when Local is used independently. The first result is in agreement with what could be expected for tf.idf in the traditional vector space for text retrieval.

As for the second result, it can be explained by the distribution of the index terms in the extended space, which differ from a standard vector space because of the expansion operation. Star graphs located in the upper part of the lattice (i.e., close to $T_{sg}$) tend to occur more often than star graphs in the lower part (i.e., close to $bot_{sg}$), since the expansion adds terms that are generic to those belonging to the index. Besides, due to the sparseness of the space, a small number of terms appear fairly often in the collection, while it is rare to see a wide range of terms with a large Global value.

6.3. Processing time and scalability

A short processing time for a query is a central requirement for interactive information systems. In IR, we distinguish between indexing processing time and search processing time. While a rather expensive indexing processing time is acceptable, it is crucial to guarantee a short search processing time. The indexing approach that we implemented requires close to $10$ s per image to generate the star graph index, including the time to generate the labels. Note that this step is performed only once, for all evaluated systems. We emphasize that besides providing a balanced representation between concepts and conceptual graphs, the proposed star graph vector space approach benefits from an important feature of the VSM that is the low processing complexity.

Indeed, with a given vocabulary size, the required evaluation time for a query is linear to the number of documents. This feature is opposed to total graph projection systems, where the NP-complexity of graph isomorphism results to non-polynomial implementations. In the context of user-oriented systems, it is an important requirement to guarantee a short processing time, whatever the size of the collection. While the average processing time for one query with DIESKAU is close to $1$ s on a desktop computer, it is three orders of magnitude shorter (about $1 \times 10^{-3}$ s per query) for both VSM-C and VSM-SG.

6.4. Robustness to labeling errors

While images from the collection Coll-1 have been annotated manually, images from Coll-2 have been indexed fully automatically. The question of the quality of annotation arises when the annotation process is automated. Indeed, the presented model assumes that an indexing process has generated labels for images beforehand, and relies on this annotation. As a consequence, the quality of the retrieval results obviously depends on the quality of the annotation. However, the comparison between evaluated systems is fair, since all systems are compared on the same basis, with the same images, same labels, and same queries. Our focus is to evaluate the difference between a standard bag-of-word search more elaborated approaches including relations.

Besides, while one could argue that the automatic document expansion is likely to amplify labeling errors, it is interesting to notice that it has no impact on the star graph model comparatively to the other models. This point shows in the experimental results, where the gain in precision for Coll-1 and Coll-2 are similar. This is explained by the fact that, on one hand, a label missing from an image (e.g. a “boat” in the image does not appear in the index) will result in this image being never retrieved by all models with this label as a query (silence). On the other hand, a wrong label (e.g. a plane is labeled with “boat”) will result in a non relevant image being retrieved by all models (noise).

6.5. Limitations of the model

Since our model is an extension of the VSM, it obviously suffers from the same limitations as the original VSM. For instance, it is not possible to express the negation of a term in a query. Therefore, a user cannot search “a boat without sea”. In addition, the model relies on an open universe assumption. For instance, the query “a tree in the center of the image” does not implicitly mean “... and no other tree in the image”. When generating the relevant document list, we considered an image to be relevant whenever there actually was a tree in the center of the image, regardless of the remainder of the image. However, it could be argued that a picture of a forest covering the whole image is not specifically relevant to this query.

Another limitation is the implicit assumption of orthogonal dimensions in the vector space, so that the use of the cosine is valid. Eq. (3) does not take into account the possible dependencies between terms, which raised criticism against this model (Raghavan & Wong, 1986). However, it has been shown that taking into account the dependencies does not allow one to improve the quality of the model: given the location of the dependencies, considering them in a global way in the collection is likely to produce the opposite effect and deteriorate the model (Baeza-Yates & Ribeiro-Neto, 1999).
7. Conclusion

In this paper, we have introduced a new framework for relational information retrieval dedicated to images. This framework includes an image model in which the physical image is segmented in a set of regions that can be aggregated to form Image Objects. Image Objects are a flexible way to delimit real-world objects in the image. When associated with concepts, they have a semantic interpretation; the relation between them is expressed via conceptual relations.

Star graphs, which are elementary expressions of the relations between concepts, are essential descriptors of the model presented. They generate a vector space, in which documents and queries are represented as vectors. The vectors coordinates denote the importance of each star graph in the index. They are estimated using a dedicated weighting scheme built on top of a model of importance of Image Objects. Along with vector expansion and similarity function, this representation provides a complete information retrieval model adapted to image retrieval.

7.1. Contributions of the paper

After implementing the model, the test results for two different collections show that:

- Document expansion yields an important gain of precision, with a ratio of about 2 for concepts, and about 2.5 for star graphs.
- Integrating the relations in the document index and the query representation is an important aspect to be taken into account in image retrieval. It helps increasing the search precision for relational queries, while not affecting non-relational ones.
- The overall retrieval performance of our system is comparable to the one of a graph projection system, while the query processing time of our system is much shorter (linear).

This vector space model of star graphs contributes to relational image indexing.

7.2. Future works

The work presented here is meant to be integrated in a fully automatic image retrieval system. Future works mainly aim at designing and testing complementary approaches to either enhance the presented model, or be compared with the current proposition.

Since the expansion process generates new dimensions in the document indexes, a side-effect is produced consisting of strong dependencies between the dimensions. This issue could be addressed by performing a theoretical and experimental analysis on the impact of increasing the dependencies. For instance, a Singular Value Decomposition of the star graph vector space may help reducing the space to useful data, as it was done before with the traditional VSM for text.

A theoretical analysis of the star graph space structure will indicate how a dedicated idf could be designed for the star graph space, taking into account both the sparseness of the space and the effect of document expansion.

Regarding the weighting issue, the presented geometrical modeling could be enriched with low-level indicators of region saliency. For instance, saliency maps (Itti, Koch, & Niebur, 1998) could be used to determine the regions and the importance of star graphs. Through this approach, the importance values of the regions could be a function of the estimated saliency value of interest points in the regions.

Appendix A. Concept type lattice of Coll-1

Root > IO
Root > Image
Root > Artificial_Objet
Root > Natural_Objet
Artificial_Objet > Other_Artificial_Objet
Artificial_Objet > Bell
Artificial_Objet > Construction
Construction > Other_Construction
Building > Pond
Construction > Building
Building > Other_Building
Building > Castel
Building > Cult_Building
Cult_Building > Other_Cult_Building
Cult_Building > Church
Cult_Building > Monastery
Appendix B. Concept type lattice of Coll-2

Root > IO
Root > symb
symb > people
people > face
face > man
man > j luc
man > uncle
face > woman
woman > wife
woman > an tie
face > child
child > so n
child > moody-boy
child > funny-boy
woman > maid
woman > blonde
woman > old-lady
face > white-hair
people > crowd
crowd > heads
crowd > audience
crowd > cyclists
crowd > kids
crowd > group

Appendix C. Relation lattice

Root > comp
Root > IsA
Root > touchborder
touchborder > touchtop
touchborder > touchbottom
touchborder > touchright
touchborder > touchleft
Root > touch
Root > right_of
Root > below_of
Root > center0x
center0x > center00
References


Please cite this article in press as: Martinet, J., et al. A relational vector space model using an advanced weighting scheme for image retrieval. Information Processing and Management (2010), doi:10.1016/j.ipm.2010.10.003


Please cite this article in press as: Martinet, J., et al. A relational vector space model using an advanced weighting scheme for image retrieval. Information Processing and Management (2010), doi:10.1016/j.ipm.2010.10.003