This paper is published in Proc. of 2nd IEEE Int'l Conf. on Multimedia and Expo, pp. 313-316. Tokyo, Japan, 2001.

THESAURUS-AIDED APPROACH FOR IMAGE BROWSING AND RETRIEVAL

Jun Yang[1], *Liu Wenyin*[2], *Hongjiang Zhang*[2], *Yueting Zhuang*[1]

[1]Department of Computer Science
Zhejiang University
Hangzhou, 310027, China
yangjun@acm.org; yzhuang@cs.zju.edu.cn

[2]Microsoft Research China
49 Zhichun Road
Beijing, 100080, China
{wyliu, hjzhang}@microsoft.com

ABSTRACT

The current trend of image retrieval is to incorporate image semantics with visual features to enhance retrieval performance. Although many approaches annotate images with keywords and process query at the semantic level, they fail to explore the full potentials of semantics. This paper proposes thesaurus-aided approaches to facilitate semantics-based access to images. The contribution of our work are two-fold: constructing a *dynamic semantic hierarchy* (DSH) which supports flexible image browsing by semantic subjects, as well as formulating a *semantic similarity metric* to improve the accuracy of semantic matching. Both approaches are seamlessly integrated into a unified framework for semantics- and feature-based image retrieval. Experiments conducted on the real-world images demonstrate the effectiveness of our approaches.

1. INTRODUCTION

Effective and efficient access to image database has recently gained much research interest with the increasing availability of digital images. In this filed, content-based image retrieval (CBIR) is devised to search images based on visual similarity [1]. However, its performance is severely limited due to the gap between image semantics and visual features. Therefore, many state-of-the-art image retrieval systems are inclined to integrate semantics (keywords) with visual features into a unified framework, allowing them to benefit each other to yield better performance.

In our previous research, we developed a prototype system *iFind* for image retrieval, which implemented a semi-automatic image annotation strategy [5] and a unified framework for semantics- and feature-based image retrieval and relevance feedback [2]. Firstly, we construct a semantic network for the images in the database. In this network each image is linked to a set of keywords relevant to its semantic content. A weight is assigned to each link to show the descriptive power of the corresponding keyword. As such image-keyword links may not be available initially, they can be obtained interactively by the semi-automatic annotation strategy [5] during relevance feedback. Whenever the user provides a set of relevant/irrelevant images to the input query keywords, an underlying voting scheme is triggered to propagate or update the image annotation: 1) For each relevant image, if it hasn't been annotated with the query keyword, create a link between them with an initial weight. Otherwise we increase the keyword weight by a given increment. 2) For each irrelevant image, if the query keyword is linked to it, decrease its weight by some degree (e.g., one fourth). In this way, the annotation is propagated/updated in a hidden manner in the course of user interaction and hence improves both the coverage and quality of the annotation among the images. We then developed a unified framework under which semantics can be seamlessly integrated with visual features for image retrieval and feedback. Experiments manifest that by using this framework higher retrieval accuracy is achievable with less iterations of feedback than using traditional relevance feedback methods [6].

However, there are also severe disadvantages with *iFind*. On the one hand, it lacks the support to image browsing/navigation by semantic subjects, which requires images being explicitly organized by their semantic subjects, instead of being implicitly annotated with keywords as is the case in *iFind*. In addition, a number of categories predefined in *iFind* are too rough and inflexible to facilitate convenient browsing. On the other hand, the semantic similarity is measured as the number of keywords in common between query and image annotation. It overlooks the similarity between different words and thus is vulnerable to the problems posed by the richness of natural language, such as synonyms, polysemy and other complex word relevancy. This drawback is especially prominent when considering users preferences on use of different keywords describing the same image.

In this paper, we present a power of thesaurus-aided approach to addressing the aforementioned problems. In particular, we utilize an electronic thesaurus WordNet [4] to interactively construct a *dynamic semantic hierarchy* (DSH) as the image categories to support flexible browsing. We also formulate a *semantic similarity metric* to improve the accuracy of semantic matching. Both methods are integrated into *iFind* [2] to enhance its performance and facility.

2. THE PROPOSED APPROACHES

WordNet [4] is an electronic thesaurus that models the lexical knowledge of English language. It is organized around the concept of synset as a class of closely related synonyms representing the same word sense. There are also various types of semantic links among synsets, which constitute a highly interconnected network of synsets. For simplicity, we apply the following rules when using WordNet: 1) we use only nouns in WordNet since nouns are more intensively used to describe images than other classes of words; 2) for a word with multiple senses, we take its first sense as the user-intended sense, so that we can use synset and word interchangeably since each word is mapped exactly to one synset; 3) among various semantic links between nouns, we use only synonyms and hypernym/hyponym relationships. These simplifications reduce WordNet to a number of synsets hierarchies going from generic concepts at higher levels to specific concepts at lower levels, which is referred to as WordNet hierarchies and exemplified in Figure 2(a).

2.1 Dynamic Semantic Hierarchy (DSH)

As keywords are accumulated semi-automatically in *iFind*, our key idea is to construct hierarchical categories from all these

keywords to support image browsing. Although WordNet itself provides well-structured and comprehensive semantic hierarchies, it is not applicable to image browsing due to its huge size (approximately 48,800 noun synsets). Therefore, we have devised the dynamic semantic hierarchy (DSH) as a set of sub-hierarchies of the WordNet hierarchies, which can be expanded interactively and progressively from predefined root concepts.

Initially, DSH is an empty hierarchy with only several root concepts available, including *action*, *living form*, *object*, *place*, *event*, *phenomenon*, *group*, *possession* and *condition.* We also create a virtual root named "everything" and insert all other root concepts as its offspring. DSH will be then expanded each time a new keyword is identified by the system, such as annotating images with new keyword, or marking images as relevant to a new word query. Such operations will cause the synset of the keyword to be inserted into the proper position in DSH. Sometimes the ancestor synset of that keyword will also be inserted. The insertion operation will always keep the structure of DSH in conformity with WordNet hierarchies. The detailed algorithm of inserting keyword *w* into DSH is given in **Figure 1.** Since a vast majority of nouns in WordNet are very infrequently occurring words (determined by their frequency in a text corpus) that should not appear in DSH, our algorithm avoids inserting them into DSH.

Step 1: *Find the corresponding synset S_n of w in WordNet.*

Step 2: *Start from S_n , trace bottom- up along the links in WordNet hierarchy, until the first ancestor synset of S_n that has already existed in DSH is reached. This ancestor synset is denoted as S_a.*

Step 3: *Set $\{S_1, S_2, \ldots, S_M\}$ as the direct children of S_a in DSH, where M is the number of children.*

For i=1 to M

Find the lowest common ancestor synset S_{co_a} of both S_i and S_n in WordNet hierarchy.

If $S_{co\_a} <> S_a$, go to Step 5.

End For

Step 4: *Insert S_n into DSH as a direct child of S_a.*

Exit the algorithm.

Step 5: *Insert S_{co_a} into DSH as a child of S_a.*

Remove S_i as the child of Sa and then insert it as a child of Sco_a.

If $S_n <> S_{co\_a}$, insert S_n into DSH as a child of Sco_a.

Exit the algorithm.

Figure 1. Insertion algorithm for DSH.

Insertion operations using this algorithm are exemplified in **Figure 2**. Figure 2 (a) is a fragment taken from the original WordNet hierarchy, with the infrequent nouns shown in dashed rectangle. Three keywords are inserted into DSH under the root concept *living form* in the order of *parrot*, *horse* and *hen*. The resultant DSH after each insertion is displayed respectively in Figure 2(b)~(d), with the keywords inserted at each step shown in bold. Obviously, DSH is semantically well-structured and compact compared with the original WordNet hierarchy.

We use the insertion of *horse* (see (c)) as an example to demonstrate how our algorithm works. Firstly, *horse* is mapped to the synset S_n (step 1) and its first existing ancestor (in DSH) S_a is located in the WordNet hierarchy, which is *living form* in this case (step 2). Since parrot is the only child of *living form* in DSH, we find its common ancestor S_{co_a} with *horse* (step 3), which is *animal*. Note that *chordate* and *vertebrate* are skipped as infrequent nouns. Since S_{co_a} is not equal to S_a, we jump the loop in Step 3 to Step 5, in which we insert *animal* under *living form* and connect both *parrot* and *horse* as its children.

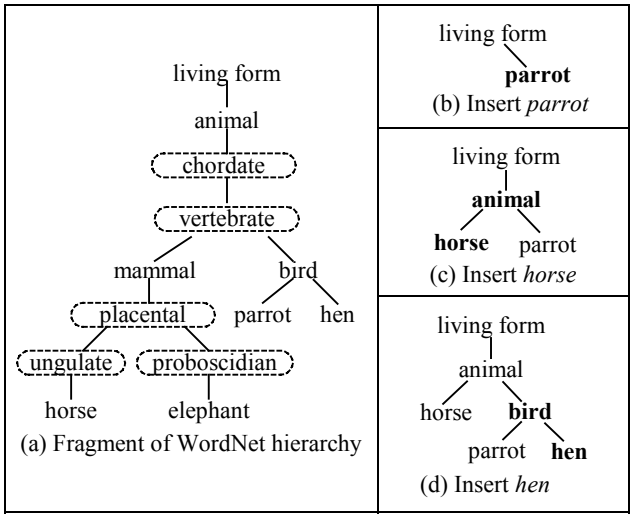

Figure 2. Illustration of insertion operations in DSH.

A likely concern of this algorithm is the loop in step 3, which will be stopped as soon as the first common ancestor of S_i and S_n that is not equal to S_a is found. One may argue that there can be more than one such common ancestor and in that case the algorithm fails to structure DSH correctly. However, this cannot actually happen because our algorithm guarantees that S_a is always the lowest common ancestor of any two children of S_a and each node has only one parent.

2.2 Semantic Similarity Metric

As stated above, *iFind* matches query with image annotations by the number of keywords in common, which we refer to as exact keyword match scheme. This scheme ignores the semantic similarity between different words and consequently fails to address the following particular issues:

1) unable to match closely related synonyms, e.g. a query of *soccer* cannot match the images labeled with *football*.

2) unable to match generic concept with its specific concepts, e.g. the query *sports* is unable to get *football* images.

3) unable to return promising candidates in case of no exact keyword match. For instance, if the query is *football*, *iFind* will return random list if no image is annotated with *football*. However, it is more reasonable to put *sports* images (if there is any) in top ranks since they are semantically closer than other random images to the query.

To address the problems, we rely on WordNet to define a quantitative semantic similarity metric and configure it to be used in *iFind*. This metric is defined in the following two steps:

• **Word-Word Similarity:** As the first step, we define the word-word similarity based on the WordNet hierarchies, which provide a thorough and domain-independent knowledge base of semantic relationships. The word-word similarity is transformed to the similarity between their corresponding synsets. The similarity between two synsets s_1 and s_2 in the same WordNet hierarchy is determined by the depth of their

lowest common ancestor synset s_a, i.e., the number of links from root to it in the hierarchy. The similarity is then normalized by dividing the maximum possible depth and thus results in [0,1], given as:

$$sim(s_1, s_2) = \frac{depth(s_a)}{depth_{\max}} \quad (1)$$

Two particular situations need to be addressed: If s_1 and s_2 is the same synset, their similarity is set to one; or if they belong to different hierarchies, their similarity is set to zero since they are too far away to have any semantic relevancy.

This similarity metric is based on the observation of WordNet as an inheritance system, in which the property of an ancestor is inherited by all of its descendants. In addition, the hierarchy goes from generic concept at higher levels to specific concept at lower levels. Therefore, the lowest common ancestor of two synsets represents their common property, with its depth as an implication of how specific such property is. The more specific their common property is, the more similar the two synsets are, and vice versa. See Figure 2 (a) as an example: Determined by the depth of the lowest common ancestor, *horse* is more similar to *elephant* than to *parrot*. This makes sense intuitively because both *horse* and *elephant* are *mammals*, whereas the property *horse* and *parrot* have in common is that they are both *animals*.

- **Query-Annotation Similarity:** The word-word similarity is then utilized to measure the similarity between query and image annotation. In *iFind*, both the query and image annotation are represented by weighted keyword sets. The image annotation is expressed as $A=\{<a_1, w_{a_1}>, \cdots, <a_m, w_{a_m}>\}$, where a_i is the keyword (synset) in image annotation with w_{a_i} being its weight. Similarly, the query is denoted as $Q=\{<q_1, w_{q_1}>, \cdots, <q_n, w_{q_n}>\}$. All keywords in the user-submitted query may have the same weights initially, but can be set differently in feedbacks, as shown later.

Here we adopt the approach suggested by Smeaton et al. [3] to extend the similarity metric between two single words to that between two word sets, as follows:

$$sim(Q, A) = \frac{\sum_{i=1}^{n} \max_{j=1..m} \{sim(q_i, a_j) \cdot w_{q_i} \cdot w_{a_j}\}}{n} \quad (2)$$

where $sim(q_i, a_j)$ is the similarity between q_i and a_j, calculated using the similarity metric in (1).

This approach finds the best-matching keyword in the image annotation for each query keyword and computes the average of these maximum similarities as the final similarity between the query and the annotated image. Incorporating keyword weight w_{a_i} and w_{q_i} into the similarity is reasonable and intuitive, since the keyword with higher weight is more descriptive and should contribute more to the similarity than other words. Note each weight is normalized into [0,1] by Gaussian normalization. Therefore, the similarity given by (2) will also result in values between [0,1].

3. IMPLEMENTATION

We have implemented the proposed approaches in our *iFind* system to improve its performance and functionalities.

3.1 Category-Based Browsing Tool

The main interface of the updated *iFind* system integrates the query interface, image browser and feedback interface together, as shown in **Figure 3.** Our proposed DSH is visualized by the tree control at the top-left pane of the interface. As its rendering tool, the tree control keeps in conformity with DSH. At first it contains only items denoting the root concepts of the DSH. Whenever a keyword is inserted into the DSH, a corresponding item will be created in the same position of tree control, representing a new category. All the images annotated with that keyword will be also put under the new category. By clicking the symbol "+" and "-" on the left-hand side of each item, users can expand/close the item to display/hide all its children categories. With the help of such hierarchical categories, users can recursively trace down the hierarchy to find out the category to his/her interest and browse all the images in this category by double-clicking the item.

Figure 3. Main user interface of the updated *iFind*.

As we can see, such DSH-driven hierarchical categories provide a more convenient means for image browsing compared with the browsing tools using predefined categories. It is capable of learning new keywords and adding them dynamically as a category into appropriate position. It is also carefully tailored to include only the used keywords and is therefore very compact.

3.2 Integrating Semantics with Visual Features

The semantic similarity metric is integrated into *iFind* as the substitution of the previous exact keyword match scheme. Hence, the keyword-based query is conducted by matching the query with the annotation of each image using the similarity metric defined in (2). The semantically matched images will be retrieved if there are any; otherwise random list is returned.

After collecting the positive and negative feedback examples, the system performs relevance feedback at the feature level using the methods suggested by Rui and Huang [6], in parallel with that at the semantic level using the semi-automatic annotation scheme [5]. The refined retrieval results after feedback will be calculated as follows: All distinct keywords ever appeared in the annotation of positive examples are collected to compose a pseudo-query Q_p, with the keyword weights set to the numbers of its occurrence of in such annotations. A similar pseudo-query Q_n is constructed from negative feedbacks. The final similarity S_i of the *ith* image in

the database to the query is calculated using the following updated semantics and visual features based on the the traditional Rocchio 's formula [7]::

$$S_i = sim(Q, A_i) + \alpha(1 + sim(Q_p, A_i)) \sum_{k \in Np} S_{ik} - \beta(1 + sim(Q_n, A_i)) \sum_{k \in Nn} S_{ik}$$

where α and β are constants, S_{ik} is the visual similarity between the ith image and the kth feedback example, N_p and N_n are the total number of positive and negative feedbacks, respectively.

4. EVALUATION

To show the advantage of our approach over the exact keyword match scheme, a particular experiment is performed on our ground-truth image database, which is constructed using 5,000 images collected from the Corel Image Gallery. These images are classified into 50 categories with 100 images in each category. Images within the same category are regarded as relevant to each other and can be described by the category name. Thus, if the category name is used as query, all the 100 images of this category are expected to be retrieved by the system.

The experiment is conducted with the help of 20 human subjects who has no knowledge on image retrieval. At first each subject was asked to browse through all the images by categories without knowing the category name. Later they were required to submit query intended to search for images from exact one category, using their own keyword that they thought might be descriptive to the category. In *iFind* there are 10% images in each category that have been annotated with the category name. For each query we examine the first 100 images ranked top in the retrieval list. Among these 100 images, the system automatically marks those belonging to the intended category as positive examples and the rest as negative ones. Hence, the system can improve the retrieval results with more relevant images. The same process is repeated in further iterations of feedback and the statistics (hit and miss) at each iteration are recorded. Since the number of images retrieved is equal to the number of relevant images, the value of precision and recall is the same and referred to as "retrieval accuracy" in this paper.

We asked each subject to submit five queries (totally 100 queries) and compared the average retrieval accuracy of our semantic similarity metric with that using exact keyword match. As shown in **Figure 4,** our approach outperforms the exact keyword match scheme by an average of 10% of accuracy.

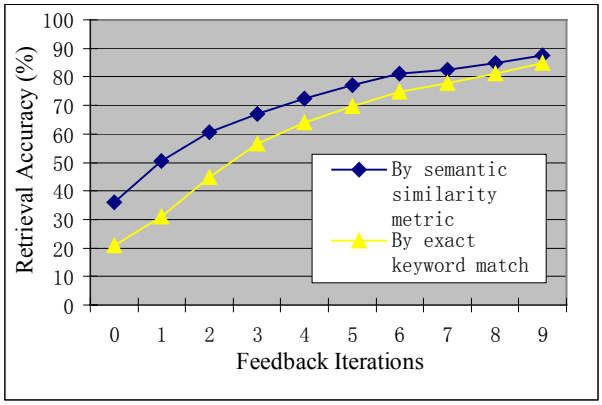

Figure 4. Performance comparison.

In the experiment, it is noticed that the probability of the query keyword used by human subject coinciding with the category name is 58%, in which case our approach reduces to the exact keyword match scheme. In the remaining queries, the keyword used is different with but closely related to category name. In this case our semantic similarity metric outperforms the exact keyword match scheme, because it can still match 10% initially annotated images while the latter cannot.

5. CONCLUDING REMARKS

In this paper, we have presened the power of thesaurus-aided approaches to support semantics-based access to image database. We construct the *dynamic semantic hierarchy* interactively and progressively from WordNet, which is then visualized in *iFind* as the hierarchical category-based image browsing tool that features flexibility and convenience. We also formulate a novel *semantic similarity metric* that outperforms the exact keyword match scheme in capturing the variety of relevant keywords used in queries. It is incorporated with visual similarity under the unified framework in *iFind* and helps it achieve a higher retrieval performance.

Currently our approaches are tailored to and incorporated into *iFind*. However, it is actually general enough to be combined with other systems, given that the image semantics is available. Since both of our approaches suffer from some ill organizations of words in WordNet, we attempt to use some other thesaurus to improve our approaches in our future work.

6. REFERENCES

[1] Rui, Y., Huang, T. S., Chang, S. F. "Image Retrieval: Current Techniques, Promising Directions and Open Issues", *Journal of Visual Communication and Image Representation*, Vol. 10, 39-62, 1999.

[2] Lu, Y., Hu, C. H., Zhu, X. Q., Zhang, H. J., Yang, Q. "A Unified Framework for Semantics and Feature Based Relevance Feedback in Image Retrieval Systems", *ACM Multimedia,* pp 31-38, 2000.

[3] Smeaton, A. F., Quigley, I. "Experiments on Using Semantic Distance Between Words in Image Caption Retrieval", *In Proc. of the 19th International Conference on Research and Development in Information Retrieval,* Zurich, Switzerland, 1996.

[4] Miller, G. A., Beckwith, R., Felbaum, C., Gross, D., Miller, K. "Introduction to WordNet: An On-line Lexical Database", *International Journal of Lexicography*, Vol.3, No.4, 235-244, 1990.

[5] Liu, W. Y., Sun, Y. F., Zhang, H. J. "MiAlbum—A System for Home Photo Management Using the Semi-Automatic Image Annotation Approach", *Technical Report,* Microsoft, 2000.

[6] Rui, Y., Huang, T. S. "A Novel Relevance Feedback Technique in Image Retrieval", *ACM Multimedia* 1999.

[7] Rocchio J.J. "Relevance Feedback in Information Retrieval", In: *The SMART Retrieval System*, pp. 313-323, Prentice Hall.