Visual Semantic Complex Network for Web Images

Shi Qiu[1], Xiaogang Wang[2,3], and Xiaoou Tang[1,3]
[1]Department of Information Engineering, [2]Department of Electronic Engineering, The Chinese University of Hong Kong
[3]Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences
{qs010, xtang}@ie.cuhk.edu   xgwang@ee.cuhk.edu.hk

Abstract

This paper proposes modeling the complex web image collections with an automatically generated graph structure called visual semantic complex network (VSCN). The nodes on this complex network are clusters of images with both visual and semantic consistency, called semantic concepts. These nodes are connected based on the visual and semantic correlations. Our VSCN with $33,240$ *concepts is generated from a collection of* 10 *million web images.* [1] *A great deal of valuable information on the structures of the web image collections can be revealed by exploring the VSCN, such as the small-world behavior, concept community, in-degree distribution, hubs, and isolated concepts. It not only helps us better understand the web image collections at a macroscopic level, but also has many important practical applications. This paper presents two application examples: content-based image retrieval and image browsing. Experimental results show that the VSCN leads to significant improvement on both the precision of image retrieval (over 200%) and user experience for image browsing.*

1. Introduction

The enormous and ever-growing amount of images on the web has inspired many important applications related to web image search, browsing, and clustering. Such applications aim to provide users with easier access to web images. An essential issue facing all these tasks is how to model the relevance of images on the web. This problem is particularly challenging due to the large diversity and complex structures of web images. Most search engines rely on textual information to index web images and measure their relevance. Such an approach has some well known drawbacks. Because of the ambiguous nature of textual description, images indexed by the same keyword may come from irrelevant concepts and exhibit large diversity on visual content. More importantly, some relevant images under different keyword indices such as "palm pixi" and "apple iphone" fail to be connected by this approach. Another approach estimates image relevance by comparing visual features extracted from image contents. Various approximate nearest neighbor (ANN) search algorithms (*e.g.* hashing) have been used to improve the search efficiency. However, such visual features and ANN algorithms are only effective for images with very similar visual content, *i.e.* near duplicate, and cannot find relevant images that have the same semantic meaning but moderate difference in visual content.

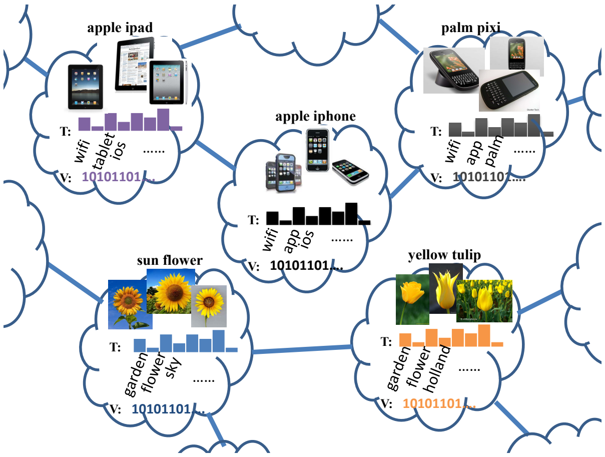


Figure 1. Illustration of the VSCN. **T** and **V** are textual and visual descriptors for a semantic concept.

Both of the above approaches only allow users to interact with the huge web image collections at a microscopic level, *i.e.* exploring images within a very small local region either in the textual or visual feature space, which limits the effective access of web images. We attribute this limitation to the lack of a top-down organization of web images that models their underlying visual and semantic structures. Although efforts have been made to manually organize portions of web images such as ImageNet [6], it is derived from a human-defined ontology that has inherent discrepancies with dynamic web images. It is also very expensive to scale.

The purpose of this work is to automatically discover and model the visual and semantic structures of web image collections, study their properties at a macroscopic level,

[1] Our VSCN data can be downloaded from **http://mmlab.ie.cuhk.edu.hk/project_VSCN.html**

and demonstrate the use of such structures and properties through concrete applications. To this end, we propose to model web image collections using the *Visual Semantic Complex Network* (VSCN), an automatically generated graph structure (illustrated in Figure 1) on which images that are relevant in both semantics and visual content are well connected and organized. Our key observation is that images on the web are not distributed randomly, but do tend to form visually and semantically compact clusters. These image clusters can be used as the elementary units for modeling the structures of web image collections. We automatically discover image clusters with both semantic and visual consistency, and treat them as nodes on the graph. We refer to the discovered image clusters as *semantic concepts*, and associate them with visual and textual descriptors. Semantic concepts are connected with edges based on their visual and semantic correlations. The semantic concepts and their correlations bring structures to web images and allow more accurate modeling of image relevance. Our VSCN currently comprises $33,240$ semantic concepts and around 10 million web images. Each concept contains an average of 300 exemplar images. Given more computation resources, this complex network can be readily scaled by including more concepts and more images under each concept.

We can better understand web image collections at a macroscopic level by studying the structural properties of the VSCN from the perspective of complex network [1]. We explore a few of them in this work, including small-world behavior, concept community, hub structures, and isolated concepts, and reveal some interesting findings. Such properties provide valuable information that opens doors for many important applications such as text or content-based web image retrieval, web image browsing, discovering popular web image topics, and defining image similarities based on structural information [22].

We devote particular attention to two applications: content-based image retrieval (CBIR) and image browsing. For web-scale CBIR, existing approaches typically match images with visual features and ANN search algorithms (*e.g*. hashing). These algorithms often lead only to a small portion of images highly similar to the query (near duplicate). In this work, these detected images are connected to other relevant images that form community structures on the VSCN. Therefore, many more relevant images can be found by exploiting the structural information provided by the VSCN. In the second application, a novel visualization scheme is proposed for web image browsing. Users can explore the web image collections by navigating the VSCN without being limited by query keywords. Our study shows that the VSCN has small-world behaviour, like many other complex networks, which indicates that most semantic concepts can reach each other by taking a short path, which enables efficient image browsing.

2. Related Work

Modeling Structure of Web Images: ImageNet [6] and Tiny Images [20] both provide large-scale hierarchical structures of images, by associating web images with/without human selection to nodes in the WordNet ontology. They both inherit the structure of WordNet, which is pre-defined by human experts and does not well capture the diverse and dynamic images on the web. In contrast, our VSCN is automatically generated from the visual and textual contents on the web, making it well-suited for tasks related to web images. Visual Synset [21] and LCSS [14] learn a set of prototypical concepts from web images, but neither of them model the correlations among concepts. Their learned concepts are used independently for image annotation tasks. ImageKB [26] obtains representative entities for web images and organizes these entities by dividing them into different categories according to an entity-category table. Our VSCN differs from ImageKB in that we organize the semantic concepts using a complex network, which provides richer information about the structures of web images, as presented in Section 4.

Content-Based Image Retrieval: Content-based image retrieval (CBIR) has been studied for years, and although remarkable progress has been made in specific areas, such as particular object retrieval [16], duplicate image retrieval [27], scalable indexing [10, 8], and image re-ranking [3, 4, 19, 24, 25, 17], the fundamental problem of finding semantically similar images remains largely unsolved. It is especially difficult for web-scale image collections. In recent years, vision researchers have tried to approach this problem from several directions, including leveraging high-level semantic attributes and signatures [5, 24, 25], fusing multiple types of visual features [7], and learning semantic-preserving image similarity [2]. Such efforts have aimed to close the semantic gap by learning more powerful features and similarities, but the large variation of visual contents make this problem extremely challenging. This challenge implies the need for a better organization method that well models the structures of web images to improve web-scale CBIR.

Image Collection Browsing: An effective browsing scheme is critical for users to access their desired images [12]. A number of browsing schemes organize images on a plane based on visual similarities [13], such that images with higher visual similarities are placed closer. These methods do not consider the underlying semantic structure, which is very important for understanding the overall content of an image collection. IGroup [23] groups images using surrounding texts and enables users to browse images by semantic clusters. However, it ignores the relationships among semantic clusters. All of these approaches are more suitable for browsing an image collection under one particular query, but not the entire web image collection.

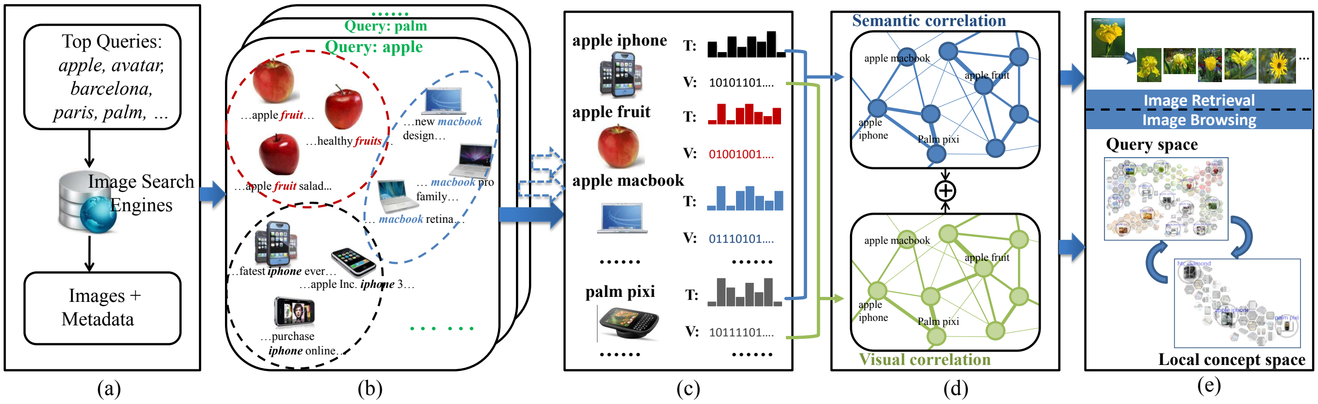


Figure 2. Flowchart of VSCN construction (best viewed in color).

3. VSCN Construction

3.1. Semantic Concept Discovery

The flowchart of constructing VSCN is shown in Figure 2. Starting with $2,000$ top query keywords of Bing image search engine, we automatically discover $33,240$ semantic concepts that are compact image clusters with visual and semantic consistency. Our method learns the semantic concepts by discovering keywords that occur frequently in visually similar images. These discovered keywords correlate well with the image content and therefore leads to descriptive concepts. We summarize the method of concept discovery in Algorithm 1. For every query q, *e.g.* “apple”, we submit q to an image search engine. With the retrieved collection of images $\mathcal{I}_q$ and surrounding texts $\mathcal{T}_q$, their relevant semantic concepts, such as “apple fruit” and “apple iphone”, can be automatically discovered. Such concepts have more specific semantic meanings and less visual diversity, and can be viewed as elementary units of web image collections. The learned concepts under query keyword q are denoted as $\mathcal{C}_q = \{c_i\}_{i=1}^{M_q}$. The concepts learned from different queries form the nodes of the VSCN.

Algorithm 1 Concept Discovery through Query Expansion

```
Input: Query $q$, image collection $\mathcal{I}_q$, surrounding texts $\mathcal{T}_q$.
Output: Learned concept set $\mathcal{C}_q = \{c_i\}_{i=1}^{M_q}$.
Initialization: $\mathcal{C}_q := \emptyset,\ r_I(w) := \mathbf{0}$.
for all images $I_k \in \mathcal{I}_q$ do
    Find the top $K$ visual neighbors, denote as $\mathcal{N}(I_k)$
    Let $W_{I_k} = \{w_{I_k}^i\}_{i=1}^T$ be the $T$ most frequent words in the surrounding texts of $\mathcal{N}(I_k)$.
    for all words $w_{I_k}^i \in W(I_k)$ do
        $r_I(w_{I_k}^i) := r_I(w_{I_k}^i) + (T - i)$.
    end for
end for
Combine $q$ and the $M_q$ words with largest $r_I(w)$ to form $\mathcal{C}_q$.
```

3.2. Inter-concept Correlations

We further explore correlations between semantic concepts. As the number of concepts is very large (33240 in this work, and potentially even larger if we expand the VSCN), we use two efficient methods to compute semantic and visual correlations as described below.

Semantic correlation is computed using the Google Kernel (GK) proposed by Sahami *et al.* [18]. We adopt this method because it has been shown to work robustly in measuring the similarity of two short texts (a short text contains a set of keywords) at the semantic level, and because of its efficiency. For a short text x, a set of Google snippets $S(x)$ is obtained from the Google web search. A Google snippet is a short text summary generated by Google for each search result item with query c. We collect the snippets of the top N search result items, which provide rich semantic context for x. We then determine the similarity between two texts x_1 and x_2 by computing the textual similarity between $S(x_1)$ and $S(x_2)$ using the term vector model and cosine similarity. For each concept $c_i \in \mathcal{C}$,

1. Use c_i as a query input on the Google web search.
2. Collect the top 50 Google snippets, denoted as $S(c_i)$.
3. Compute the term frequency (TF) vector of $S(c_i)$ and keep the top 100 terms with highest TFs.
4. L_2-normalize the truncated vector, and denote the result vector as $ntf(c_i)$.

The semantic correlation between c_i and c_j is:

$$S\_Cor = Cosine(ntf(c_i), ntf(c_j)). \qquad (1)$$

Visual correlation of two concepts is measured by the visual similarity between their corresponding exemplar image sets. For each concept, its exemplar image set consists of the top 300 images retrieved from the search engine by using the concept as query keyword. This exemplar image set is further represented as a binary code by sim-hashing algorithm [15]. This sim-hashing code can be viewed as a visual signature of the original exemplar image set. The visual similarity between any pair of exemplar image sets can then be approximated by the negative of hamming distance between their sim-hashing codes. Concretely, for a concept $c_i \in \mathcal{C}$, we denote its exemplar image set by $\mathcal{I}_{ci}$.

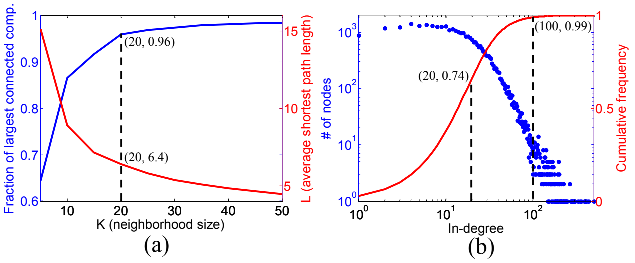


Figure 3. (a) Size and the average shortest path length of the largest connected component on the VSCN. (b) In-degree distribution and cumulative frequency. See Section 4.1 and 4.2 for details.

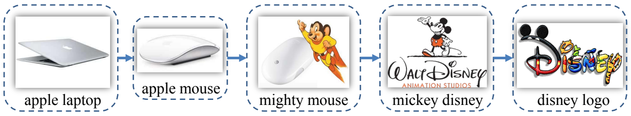


Figure 4. The path connecting two semantically irrelevant concepts "apple laptop" and "disney logo". "mighty mouse" on the path is a key step that contains two dominant components—computer mouse and cartoon character—from different domains.

1. $I_k \in \mathcal{I}_{ci}$ is encoded in an M-dimensional binary vector $\boldsymbol{H}(I_k)$ using an M-bit base hashing function $\boldsymbol{H}$ [2].
2. Accumulate the binary vectors as $\boldsymbol{A} = \sum \boldsymbol{H}(I_k)$.
3. Quantize the accumulated vector back to binary vector, $simhash(c_i) = sign(\boldsymbol{A})$.

The visual correlation between c_i and c_j is,

$$V\_Cor = 1 - \frac{1}{M} HamDist(simhash(c_i), simhash(c_j)).$$

We fuse the semantic correlation and visual correlation by $Cor = S\_cor + V\_cor$. Finally, we build the VSCN as a K-nearest-neighbor (K-NN) graph by connecting each node to its top K neighbors with the largest correlations.

3.3. Complexity

After downloading the images and metadata, our method takes 70 seconds to learn semantic concepts from one query. Discovering all 33,240 concepts takes 40 CPU hours. The inter-concept correlations requires the computation of cosine similarity between sparse word histograms and hamming distance between binary vectors, both of which can be done efficiently. Computing the two types of correlations takes 3 and 11 CPU hours, respectively.

4. Exploring Structures of the VSCN

Complex networks have many important properties [1], some of which are explored with our VSCN in this section. The study of these properties not only yields a better understanding of web image collections at a macroscopic level, but also provides valuable information that assists in important tasks including CBIR and image browsing, as presented in Section 5 and 6.

[2] Here we represent each bit with ±1.

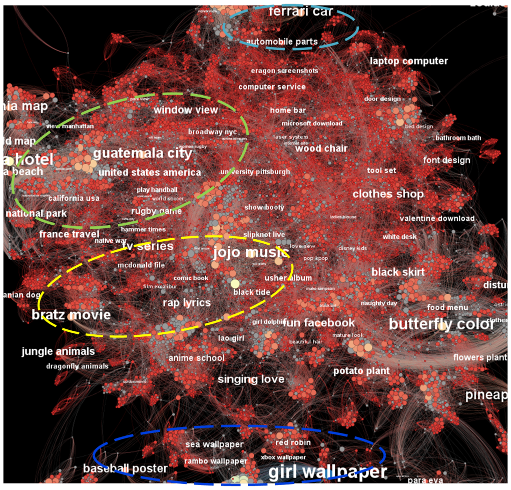


Figure 5. Part of the VSCN. Ellipses indicate different semantic regions. See Section 4.2 for details.

4.1. Small-World Behavior

The small-world behavior exists in many complex networks such as social networks and the World Wide Web. It means that most nodes can be reached from the others in a small number of hops. It is of great interest to study whether this phenomenon also exists in our VSCN. The small-world behavior has important implications in some applications such as image browsing by navigating the VSCN.

As the VSCN is constructed locally, it is interesting to know how it is globally connected. We find that even for a small neighborhood size ($K = 5$), there already emerges a dominant connected component that includes more than half of the nodes on the VSCN, as shown in Figure 3 (a). The largest connected component grows quickly with K and covers 96% of the VSCN when $K = 20$. Thus, the VSCN is a well connected network.

We compute the average shortest path length [1] by

$$L = \frac{1}{|V|(|V|-1)} \sum_{v_i, v_j \in V, v_i \neq v_j} d(v_i, v_j). \qquad (2)$$

V is defined as the largest connected component to avoid divergence of L. Figure 3 (a) shows L as a function of K. L drops quickly at the beginning. For $K > 20$, the average separation between two nodes on V is only about six hops. The existence of a dominant connected component and its small separation between nodes suggest it is possible to navigate the VSCN by following its edges, which inspires the novel image browsing scheme introduced in Section 6. In the rest of this paper, K will be fixed at 20. It is interesting to see how semantically different concepts are connected on the VSCN as exemplified in Figure 4.

4.2. In-degree Distribution

In-degree is an important measurement in complex networks. On the VSCN, the nodes have identical out-degree ($K = 20$), but their in-degrees differ widely from 0 to 500,

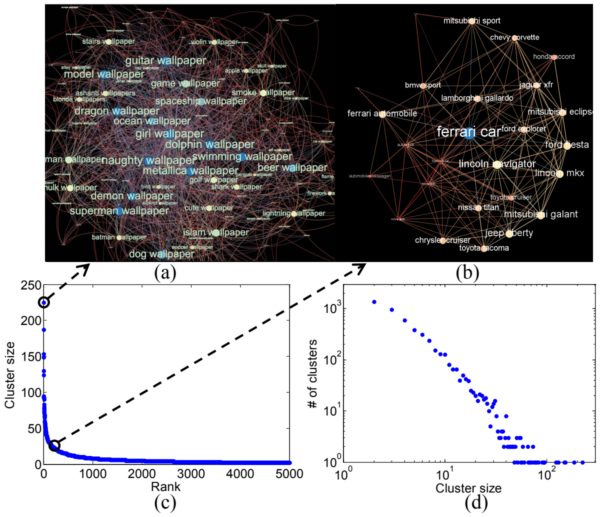


Figure 6. (a) and (b) Two communities, *wallpaper* and *sports cars*, on the VSCN. (c) Rank of cluster size. (d) Distribution of cluster size. See Section 4.3 for details.

and the distribution of in-degrees are highly skewed (Figure 3 (b)). The low in-degree part (in-degree less than 20) is close to an uniform distribution, while the high in-degree part approximates a power-law distribution. The cumulative frequency of in-degrees shows that 74% of nodes have in-degrees less than 20. Only 1% of nodes have in-degrees larger than 100. In general, representative and popular concepts that are neighbors of many other concepts have high in-degrees, and form hub structures. Isolated concepts have zero in-degree. They are typically uncommon concepts such as “geodesic dome”and “ant grasshopper”, or the failures of concept detection such as “dscn jpg” which does not have semantic meanings. Figure 5 shows part of the VSCN, with concepts of large in-degrees. We can identify several semantic regions formed by these concepts, including traveling, entertainments, wallpapers, and automobile, which correspond to the green, yellow, dark blue, and light blue regions, respectively.

4.3. Concept Community

The semantic regions observed from Figure 5 suggest the existence of community structures on the VSCN. In the literature of complex networks, a community is referred to as a subgraph with tightly connected nodes. On the VSCN, it corresponds to a group of closely related semantic concepts, called a concept community. To find such communities, we adopt the graph-based agglomerative algorithm in [28] due to its good performance and high efficiency. The algorithm starts by treating each single node as a cluster, and iteratively merges clusters with largest affinity, measured via the product of in-degrees and out-degrees between the two clusters. We cluster the nodes on the VSCN into $5,000$ groups.

We observe a few interesting facts from the clustering results. First, the size of clusters approximately follows a power-law distribution (see 6 (d)), and 10% of the clusters are with size larger than 10. They cover 52% nodes on the VSCN. Second, these clusters correspond to various semantic topics, such as cars, food, plants, and animals. Figure 6 (a) and (b) show the structures of two exemplar clusters. Figure 6 (a) shows a concept community related to “wallpaper”, which has 225 concepts. Figure 6 (b) shows another community with a moderate size, which can be interpreted as a topic related to “sports cars”.

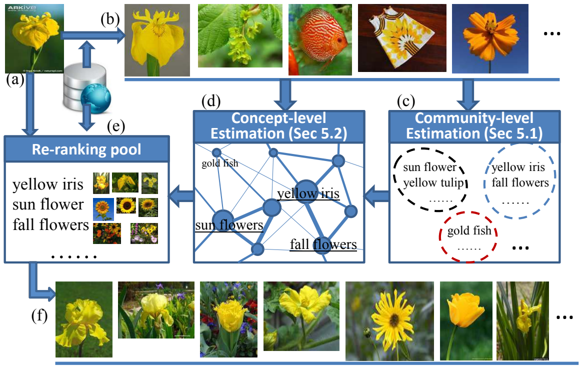


Figure 7. Flowchart of our VSCN-based image retrieval.

5. CBIR with the VSCN

In this section, we show that the VSCN is able to substantially improve the performance of CBIR systems. The key idea is to effectively reduce the search space by exploiting the structures of web images encoded in the VSCN. Our approach is illustrated in Figure 7. Given a query image (Figure 7 (a)), its nearest neighbors in the database are retrieved with a baseline method (*e.g*. ITQ hashing [8]) (Figure 7 (b)). Based on the initial retrieval result, the semantic meaning of the query image is estimated using a small set of relevant semantic concepts on the VSCN (Figure 7 (c) and (d)). Images under these semantic concepts are then gathered to form a re-ranking pool (Figure 7 (e)). Images inside the pool are ranked based on their visual similarity to the query image, and the ranking list is returned (Figure 7 (f)). The VSCN brings two key benefits: (1) as the search space is greatly reduced, the re-ranking pool contains significantly less noise than the entire database, leading to superior retrieval result. (2) The re-ranking pool contains a more manageable number of images than the entire database (a few thousand v.s. millions). It allows the use of more powerful features and similarity measures, further promoting the performance.

A key step of our approach is to estimate the semantic meaning of the query image, which is done at two levels. At the community level, we estimate the query image’s semantic meaning using a set of concept communities discovered in Section 4.3. As concept communities group similar concepts, estimating the relevant communities is more reliable than estimating individual concepts. Then, at the concept level, a smaller set of relevant concepts are further identified from the previously identified communities. Both levels fully exploit the structural information of the VSCN, which makes our approach more robust.

5.1. Community-level Estimation

We denote detected concept communities by $\{T_i\}_{i=1}^{K_T}$. Given a query image I_q, a list of top-ranked images and their distances to I_q are returned by an off-the-shelf retrieval algorithm (*e.g.* ITQ hasing [8]). From the truncated list $\{(I_k, d_k)\}_{k=1}^{N_I}$, we calculate a relevance score for each T_i as:

$$s(T_i) = \sum_{k=1}^{N_I} exp(\frac{-d}{\sigma}) \cdot \chi[c(I_k), T_i]. \tag{3}$$

$c(I_k)$ is the concept to which the database image I_k belongs. $\chi[c(I_k), T_i]$ is 1 if $c(I_k) \in T_i$ and 0 otherwise. $\sigma = \frac{1}{N_I} \sum_{i=1}^{N} d_i$. After the relevance scores are calculated for all the communities, the top N_T with the largest relevance scores are kept. The union of concepts included in these concept communities is denoted by $C' = \{c'_i\}_{i=1}^{N_C}$.

5.2. Concept-level Estimation

The results of community-level estimation enable us to focus on a small subset of concepts C'. In order to best identify the most relevant concepts out of C', we jointly leverage two sources of information. The first source is the relevance score derived from the ranking list returned by the baseline algorithm. Similar to Section 5.1, we compute the initial relevance score for each concept $c'_i \in C'$ as:

$$s(c'_i) = \sum_{k=1}^{N_I} exp(\frac{-d}{\sigma}) \cdot \mathbf{1}[c(I_k) = c'_i], \tag{4}$$

where $\mathbf{1}[\cdot]$ is the indicator function, and σ is the same as that in Equation 3. As $s(c'_i)$ is not sufficiently reliable, we introduce the second source of information—correlations between semantic concepts—to refine the noisy relevance score. To this end, we further construct a graph $G'(V', E', W')$ by extracting a subgraph from the VSCN, where V' are nodes corresponding to C', E' are edges with both nodes in V', and W' are the weights associated with E'. To integrate the two information sources, we conduct a Random Walk with Restart (RWR) on G', characterized by

$$p^{n+1} = \alpha \mathbf{P}^T p^n + (1 - \alpha)\pi, \tag{5}$$

where p^n is the walker's probability distribution over V' at step n. $\mathbf{P}$ is the transition matrix derived from W' and $\pi(i) = s(c'_i) / \sum_i s(c'_i)$. The physical meaning of Equation 5 can be interpreted as, at each step, the random walker either walks, with probability α, along the E' according to the transition matrix $\mathbf{P}$ or restarts, with probability $1 - \alpha$, from a fixed probability distribution π. Therefore, the two information sources, incorporated into the two terms on the r.h.s. of Equation 5, respectively, are fused by RWR up to the balance factor α. The equilibrium distribution p of the RWR is known as the personalized PageRank vector [11], which has the following analytical solution:

$$p = (1 - \alpha)(\mathbf{I} - \alpha \mathbf{P}^T)^{-1}\pi, \tag{6}$$

where a larger probability in p indicates higher relevance of the corresponding node. We rank the semantic concepts according to their probability values in p, and reserve the top N_C to represent the semantic meaning of the query image.

Images of the top N_C concepts are gathered and form an re-ranking pool, which are matched with the query image.

5.3. Implementation Details

Multiple visual features, including Color Signature, Color Spatialet, Wavelet, EOH, HOG, and Gist, are concatenated [19]. We apply ITQ hashing [8] to compress original features into 128-bit vectors. We use ITQ hashing as the baseline as it has been shown to achieve state-of-the-art performance. At the re-ranking stage of our approach, original image features are used to generate the final ranking list. We set the parameters N_I, N_T, and N_C to $200, 5$, and 10, respectively, by a pilot experiment on a small set of query images. The balance factor α of RWR is fixed at 0.85 as recommended in [11].

5.4. Experiments of CBIR

Dataset. We collect a set of query images to search against images of the VSCN. Since the VSCN images are gathered from Bing, we collect query images from Google. We submit the names of semantic concepts to Google and obtain the top five images returned. They are combined to form a query dataset with 160K images. Images sampled from this dataset are queried against the VSCN images.

Evaluation. For each query image, the top 100 images are retrieved and are manually labelled as being relevant/irrelevant to the query image. The performance is evaluated using average precision (AP@k). As the labeling of retrieval results is labor-intensive, we sampled 10K images from the query dataset for quantitative evaluation. We compare our approach with ITQ hasing [8].

Results. Figure 8(a) shows that our approach significantly improves ITQ hashing [8] by enhancing the average top 100 precision (AP@100) from 27.0% to 51.1% (a relative improvement of 89%). Since our approach builds on the baseline method (ITQ hashing), it is important to know whether it is able to make improvement in extreme cases when the baseline performance is very low or very high. We build two smaller datasets with the most difficult and easiest query images, respectively. The difficult dataset contains 20% of the data with the lowest baseline performance in terms of AP@100, while the easy dataset contains the 20% with the highest baseline performance. The results are given in Figure 8(b). Our approach still outperforms the baseline by a large margin. Notably, on the difficult dataset, our approach boosts the AP@100 from 7.0% to 24.3% (a relative improvement of 250%). The results clearly show that the VSCN leads to huge improvement on CBIR. Exemplary retrieval results can be found on our project page.

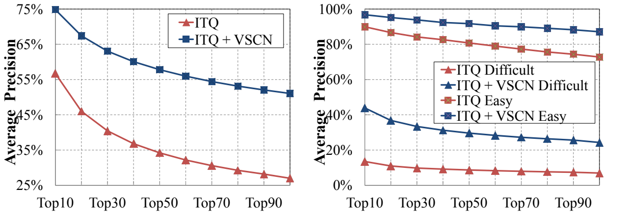


Figure 8. Retrieval performance of our approach (ITQ hasing + VSCN) and the baseline method (ITQ hasing). (a) Average precision on the 10K query dataset. (b) Average precision on the difficult and easy datasets.

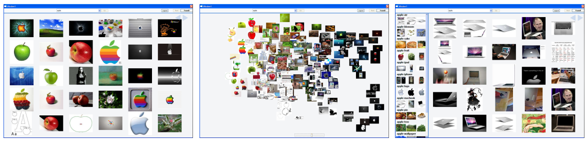
6. Image Browsing with the VSCN

This section presents a new browsing scheme that helps users explore the VSCN and find images of interest. The user starts browsing by entering a query keyword to the system. Since the size of the VSCN is huge, we provide local views. As shown in Figure 2(e), our scheme allows users to browse two spaces—the query space and the local concept space—each of which only presents a small subgraph of the entire VSCN. A query space visualizes semantic concepts generated by the same query. For example, the query space of "apple" contains concepts such as "apple fruit", "apple iphone", "apple pie", and their corresponding images. A local concept space visualizes a centric concept (*e.g.*, "apple iphone") together with its neighbor concepts (*e.g.* "htc diamond" and "palm pixi"), which may come from different query keywords. In this way, it bridges images of most related concepts and helps users access more images of interest without being limited by their initial queries.

In the browsing process, users can freely switch between the two spaces. A user who chooses a particular concept in the query space enters into the local concept space and the chosen concept becomes the centric concept. The user can then move to a new concept space by choosing a neighboring concept. In this way, users can navigate over the VSCN and search for target images. Figure 11 illustrates an image browsing process across the two spaces.

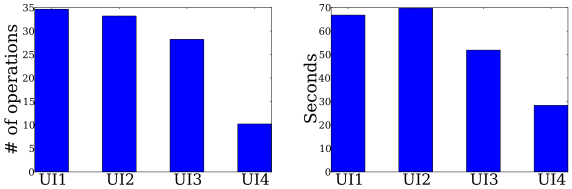
6.1. Visualizing the VSCN

Good visualization is essential for enhancing users' experience. Here, we provide an intuitive and informative method to visualize the two spaces. The subgraph in the current space is first visualized as nodes and edges. This step provides the concept-level visualization and defines the global layout of the visualization result. In image-level visualization, we present images in a hexagon lattice. Exemplar images are assigned either to cells around nodes to represent specific concepts, or to cells along edges to reflect visual transitions between concepts. The final visualization result can effectively deliver the visual and semantic content of the current space. The detailed algorithm of visualizing the VSCN is omitted here due to space limitation. [3]



(a)UI1 (b) UI2 (c) UI3

Figure 9. User interfaces compared in the user study.



(a) Users' effort (b) Search time

Figure 10. Results of user study.

6.2. User Study

We evaluate our browsing scheme by comparing it with three existing browsing schemes (interfaces): the traditional ranked-list interface, the interface of presenting images based on visual similarity [13], and the semantic cluster-based interface [23], as shown in Figure 9. We refer to the three interfaces as UI1 to UI3, respectively, and ours as UI4.

Data and Subjects. We recruit 12 subjects with image search experience to take part in the user study. We sample a subset of 20 query keywords from the VSCN. Four of them are used as examples to teach subjects how to use the four schemes. The other 16 queries are used in the task below.

Tasks. Users are asked to perform multiple rounds of search with each of the four schemes. In each round, users are first shown an image randomly sampled from the dataset and then asked to find the target image or one that they believe is close enough. Users will start from a random one of the 16 queries, and the target image is sampled from another query that is different from, yet related to the starting query. This task is designed to mimic the common scenario in which a user may not know the exact query keyword for an object and starts from another related keyword that he/she is familiar with. We allow users to reformulate query keywords as they need. The user starts/ends the search by clicking the Start/Found button, and all of the operations in between, including mouse clicks, mouse movements, and scrolling, are recorded for later analysis. Each user completes all the 16 queries with four queries assigned to each scheme. The testing order of the four interfaces is rotated for different users to reduce any possible biases.

6.3. Results

Two objective measures, *i.e.* users' effort and time consumption, are computed and analyzed using ANOVA [9].

Users' effort is measured using the average number of users' operations in the searching process, including going to next/previous page, dragging slide bars, entering/leaving clusters, switching views, and changing query keywords. Figure 10 (a) shows the average number of operations using the four schemes. It indicates that our scheme (UI4) re-

[3] The algorithm details and a video demonstration of our browsing scheme can be found on our project page.

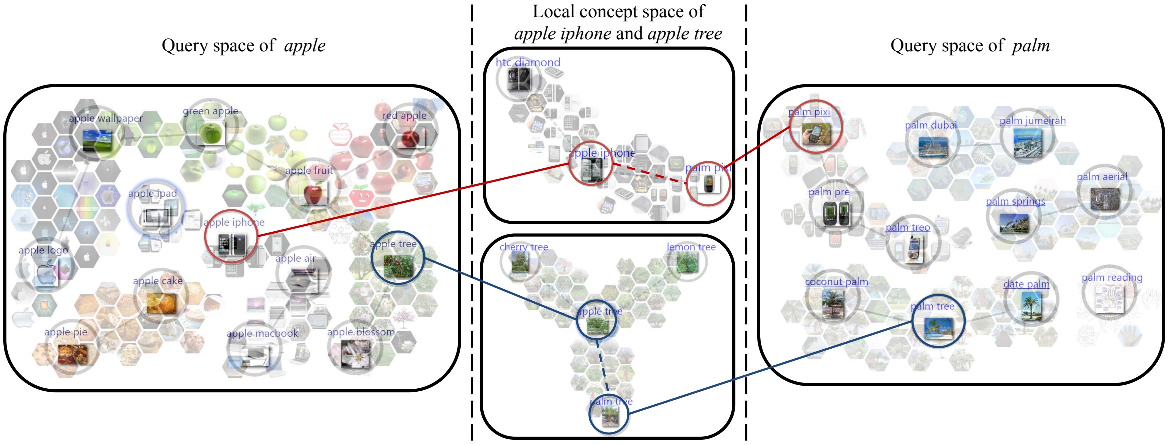


Figure 11. across query spaces and local concept spaces. Two browsing paths connecting the query spaces of *apple* and *palm* are highlighted. When users click *apple iphone* in the query space of *apple*, the local concept space is shown, with two more neighboring concepts, namely *htc diamond* and *palm pixi*. Exemplar images and visual transitions (indicated by red dashed lines) are also displayed. Users can further enter the query space of *palm* by clicking on the concept of *palm pixi*. The case is similar if users click *apple tree*.

quires the least amout of users' effort out of all the schemes. ANOVA test shows that the advantage of our scheme is statistically significant, $F(3, 212) = 15.9, p < 0.001$[4].

Average search time is a direct measure of the efficiency of the four schemes. Figure 10 (b) shows that our scheme takes the least search time, $F(3, 212) = 18.3, p < 0.001$.

7. Conclusions

This paper has proposed a novel visual semantic complex network to model the complex structures of a web image collection. We studied multiple fundamental structures of complex networks, which reveal some interesting facts about the VSCN. They not only help us understand the huge web image collection at a macroscopic level, but are also valuable in practical applications. Two exemplar applications show that exploiting structural information of the VSCN not only substantially improves the precisions of CBIR, but also greatly enhances the user experience in web image search and browsing. Many more applications of the VSCN are to be studied in future work.

Acknowledgements

The authors would like to thank Deli Zhao for helpful discussions and Wei Li for help with ANOVA. This work is supported by the General Research Fund sponsored by the Research Grants Council of the Kong Kong SAR (Project No. CUHK 416713) and Guangdong Innovative Research Team Program (No.201001D0104648280).

References

[1] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang. Complex networks: Structure and dynamics. *Physics Reports*, (45), 2006.

[2] G. Chechik, V. Sharma, U. Shalit, and S. Bengio. An online algorithm for large scale image similarity learning. In *Proc. NIPS*, 2009.

[3] J. Cui, F. Wen, and X. Tang. Intentsearch: interactive on-line image search re-ranking. In *Proc. ACM MM*, 2008.

[4] J. Cui, F. Wen, and X. Tang. Real time google and live image search re-ranking. In *Proc. ACM MM*, 2008.

[5] J. Deng, A. Berg, and L. Fei-Fei. Hierarchical semantic indexing for large scale image retrieval. In *Proc. CVPR*, 2011.

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proc. CVPR*, 2009.

[7] M. Douze, A. Ramisa, and C. Schmid. Combining attributes and fisher vectors for efficient image retrieval. In *Proc. CVPR*, 2011.

[8] Y. Gong and S. Lazebnik. Iterative quantization: A procrustean approach to learning binary codes. In *Proc. CVPR*, 2011.

[9] D. Howell. *Statistical methods for psychology*. Wadsworth Pub Co, 2009.

[10] H. Jégou, M. Douze, C. Schmid, and P. Pérez. Aggregating local descriptors into a compact image representation. In *Proc. CVPR*, 2010.

[11] A. Langville and C. Meyer. Deeper inside pagerank. *Internet Mathematics*, 1:335–380, 2004.

[12] D. Lewandowski. Search engine user behaviour: How can users be guided to quality content? *Information Sevices & Use*, 2008(28), 2008.

[13] H. Liu, X. Xie, X. Tang, Z.-W. Li, and W.-Y. Ma. Effective browsing of web image search results. In *Proc. ACM MIR*, 2004.

[14] Y. Lu, L. Zhang, J. Liu, and Q. Tian. Constructing concept lexica with small semantic gaps. *TMM*, 2010.

[15] G. Manku, A. Jain, and A. Das Sarma. Detecting near-duplicates for web crawling. In *Proc. WWW*, 2007.

[16] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. In *Proc. CVPR*, 2006.

[17] S. Qiu, X. Wang, and X. Tang. Anchor concept graph distance for web image re-ranking. In *Proc. ACM MM*, 2013.

[18] M. Sahami and T. D. Heilman. A web-based kernel function for measuring the similarity of short text snippets. In *Proc. WWW*, 2006.

[19] X. Tang, K. Liu, J. Cui, F. Wen, and X. Wang. Intentsearch: Capturing user intention for one-click internet image search. *TPAMI*, 2012.

[20] A. Torralba, R. Fergus, and W. T. Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. *TPAMI*, 2008.

[21] D. Tsai, Y. Jing, Y. Liu, H. Rowley, S. Ioffe, and J. Rehg. Large-scale image annotation using visual synset. In *Proc. ICCV*, 2011.

[22] N. Verma, D. Mahajan, S. Sellamanickam, and V. Nair. Learning hierarchical similarity metrics. In *Proc. CVPR*, 2012.

[23] S. Wang, F. Jing, J. He, Q. Du, and L. Zhang. Igroup: presenting web image search results in semantic clusters. In *Proc. ACM SIGCHI*, 2007.

[24] X. Wang, K. Liu, and X. Tang. Query-specific visual semantic spaces for web image re-ranking. In *Proc. CVPR*, 2011.

[25] X. Wang, S. Qiu, K. Liu, and X. Tang. Web image re-ranking using query-specific semantic signatures. *TPAMI*, 2013.

[26] X.-J. Wang, Z. Xu, L. Zhang, C. Liu, and Y. Rui. Towards indexing representative images on the web. In *Proc. ACM MM*, 2012.

[27] Z. Wu, Q. Ke, M. Isard, and J. Sun. Bundling features for large scale partial-duplicate web image search. In *Proc. CVPR*, 2009.

[28] W. Zhang, X. Wang, D. Zhao, and X. Tang. Graph degree linkage: Agglomerative clustering on a directed graph. In *Proc. ECCV*, 2012.

[4] In ANOVA, a smaller p-value indicates larger statistical significance. Normally, $p < 0.01$ is considered significant.