PRISM: Concept-preserving Social Image Search Results Summarization

Boon-Siew Seah
seah0097@ntu.edu.sg

Sourav S Bhowmick
assourav@ntu.edu.sg

Aixin Sun
axsun@ntu.edu.sg

School of Computer Engineering, Nanyang Technological University, Singapore

ABSTRACT

Most existing *tag-based* social image search engines present search results as a ranked list of images, which cannot be consumed by users in a natural and intuitive manner. In this paper, we present a novel *concept-preserving* image search results summarization algorithm named PRISM. PRISM exploits both visual features and tags of the search results to generate high quality *summary*, which not only breaks the results into *visually* and *semantically coherent* clusters but it also maximizes the *coverage* of the summary w.r.t the original search results. It first constructs a *visual similarity graph* where the nodes are images in the search results and the edges represent *visual similarities* between pairs of images. This graph is *optimally decomposed* and *compressed* into a set of *concept-preserving subgraphs* based on a set of *summarization objectives*. Images in a concept-preserving subgraph are visually and semantically cohesive and are described by a minimal set of tags or concepts. Lastly, one or more exemplar images from each subgraph is selected to form the *exemplar summary* of the result set. Through empirical study, we demonstrate the effectiveness of PRISM against state-of-the-art image summarization and clustering algorithms.

Categories and Subject Descriptors

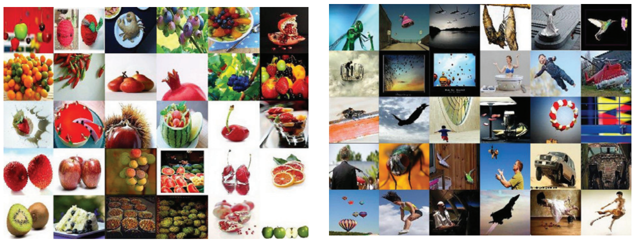
H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

Keywords

Image Search Summarization; Tag-based Image Search; Flickr

1. INTRODUCTION

The rising prominence of image sharing platforms like `Flickr` and `Instagram` has led to an explosion of social images. Consequently, the need for superior social image search engines to support efficient and effective *tag-based* image retrieval (TagIR) has become increasingly pertinent. Queries in a tag-based social image search engine are often short and ambiguous. As a result, search engines often diversify the search results to match all possible aspects of a query in order to minimize the risk of completely missing out a user's search intent [18]. An immediate aftermath of such results diversification strategy is that often the search results are not semantically or visually coherent. For example, the results of a search query "fruit" may include images of strawberries, apples, oranges, and even fruit-related concepts such as market and fruit juice as illustrated in Figure 1(a). Similarly, consider Figure 1(b) which depicts results of the query "fly". Observe that the results contain a medley of visually and semantically distinct objects and scenes (hereafter collectively referred to as *concepts*) such as parachutes, aeroplanes, insects, birds, and even the act of jumping.



(a) query: "fruit"   (b) query: "fly"

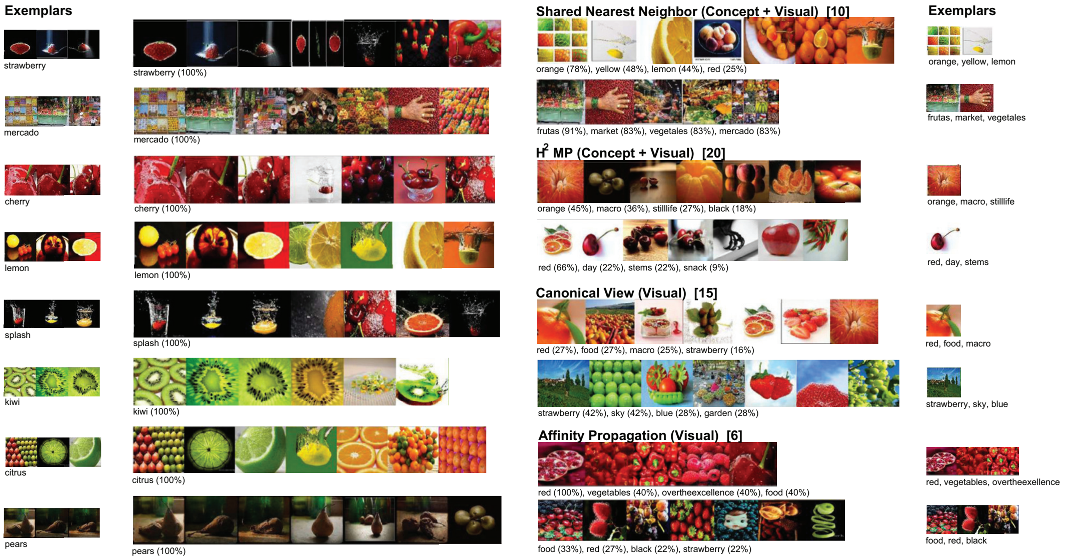
Figure 1: [Best viewed in color] Sample query results.

Image search results are typically presented as a ranked list of images often in the form of thumbnails (*e.g.,* Figure 1). Such thumbnail view suffers from two key limitations. First, it fails to provide a view of common visual objects or scenes *collectively*. For example, the result images of "fruit" and "fly" queries can be clustered by visual objects (*e.g.,* strawberry, aeroplane, insect) and activities (*e.g.,* jump). Such organized image search results will naturally enable a user to quickly identify and zoom into a subset of results that is most relevant to her query intent. Second, a thumbnail view fails to provide a bird eye view of different concepts present in a query results. For instance, reconsider Figure 1(b) containing a medley of concepts. It will be beneficial to users if a suitable exemplar image from each type of concept can be selected to create a "summary" of the search results. In this paper, we take a systematic step towards addressing these limitations associated with social image search results.

An appealing way to organize social image search results of a search query is to generate a set of *image clusters* from them such that images in each cluster are *semantically and visually coherent* and the clusters *maximally cover* the entire result set. Subsequently, at least one exemplar image from each cluster can be selected to generate an *exemplar summary* of the entire result set to give a bird's-eye view of different concepts in it. We advocate that such image clusters must satisfy the following desirable features.

- *Concept-preserving*. Each cluster should be annotated by a *minimal* set of tags generated from the images within to semanti-

SIGIR'14, July 6–11, 2014, Gold Coast, Queensland, Australia.
Copyright is held by the owner/author(s). Publication rights licensed to ACM.
ACM 978-1-4503-2257-7/14/07 ...$15.00.
http://dx.doi.org/10.1145/2600428.2609586.


Exemplars
strawberry
strawberry (100%)
mercado
mercado (100%)
cherry
cherry (100%)
lemon
lemon (100%)
splash
splash (100%)
kiwi
kiwi (100%)
citrus
citrus (100%)
pears
pears (100%)
Shared Nearest Neighbor (Concept + Visual) [10]
orange (78%), yellow (48%), lemon (44%), red (25%)
frutas (91%), market (83%), vegetales (83%), mercado (83%)
H^2 MP (Concept + Visual) [20]
orange (45%), macro (36%), stilllife (27%), black (18%)
red (66%), day (22%), stems (22%), snack (9%)
Canonical View (Visual) [15]
red (27%), food (27%), macro (25%), strawberry (16%)
strawberry (42%), sky (42%), blue (28%), garden (28%)
Affinity Propagation (Visual) [6]
red (100%), vegetables (40%), overtheexcellence (40%), food (40%)
food (33%), red (27%), black (22%), strawberry (22%)
Exemplars
orange, yellow, lemon
frutas, market, vegetales
orange, macro, stilllife
red, day, stems
red, food, macro
strawberry, sky, blue
red, vegetables, overtheexellence
food, red, black

a) Concept-Preserving Summarization (PRISM)

b) Non-Concept-Preserving Summarization

Figure 2: [Best viewed in color] Sample summarizations from 5 methods for the query “`fruit`”. The percentage associated with each tag represents the percentage of images in the cluster having the specific tag. The types of features used by a method is indicated after the method name in parentheses. A summary is constructed by selecting 1 to 3 images per cluster as exemplar.

cally[1] describe *all* images in the cluster. Users therefore can easily associate the tag(s) with the images in a cluster at a glance. We refer to such a cluster as *concept-preserving* where a set of images shares at least one concept (tag)[2]. Figure 2 depicts the distinction between concept-preserving (Figure 2(a)) and non-concept-preserving clusters (Figure 2(b)). In the concept-preserving “pear” cluster, a single “pear” tag is sufficient to represent all images in it and describe them semantically. In contrast, the “orange, yellow, lemon, red” cluster requires four tags to represent all images and furthermore the content of these images is unclear from the tags. Intuitively, it is easier to quickly digest the concept associated with the former cluster compared to the latter due to the cognitive burden imposed by multiple tags (concepts) even if these tags are related.

- *Visual coherence*. Images in a cluster must be visually coherent. Visually similar images must be clustered together and dissimilar images must be separated in different clusters.
- *Coverage*. The image clusters should cover as much of the result set as possible for maximizing incorporation of all possible query intent. In other words, image clusters should represent majority of the original result images. For instance, reconsider the set of image clusters in Figure 2(a). Assume that the “splash” and “cherry” clusters are missing. In this case, the image clusters are considered to be less complete than expected as their coverage is not maximized. Obviously, this will lead to an exemplar summary that does not *maximally cover* the result images.

Recently, *early fusion* [13, 20] and *late fusion* [10] approaches have attempted to summarize image search results. The former exploits the tags and visual content of the images jointly whereas the latter considers them independently. However, these techniques do not ensure that the generated summaries are concept-preserving and maximally covers the image results. To illustrate this, consider the “orange, macro, stilllife, black” cluster and exemplar summary in Figure 2(b) generated by [20]. With no single tag representing anywhere near 100% of the images, all the four tags are needed to describe the images in the cluster. An alternative representation is to select the “best” tag (*e.g.,* the most probable tag for a given cluster [15]). However, the “best” tag often fails to represent *all* images in the cluster, which may mislead and confuse users. For instance, consider the “strawberry, sky, blue, garden” cluster generated by [15] where no single tag can correctly represent all images.

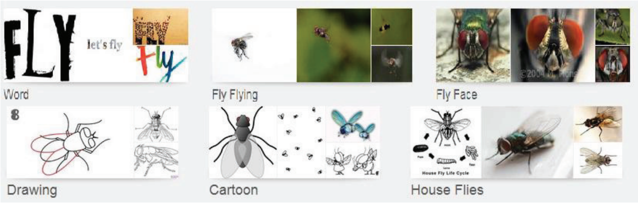


Figure 3: [Best viewed in color] *Google Images* results (“`fly`”).

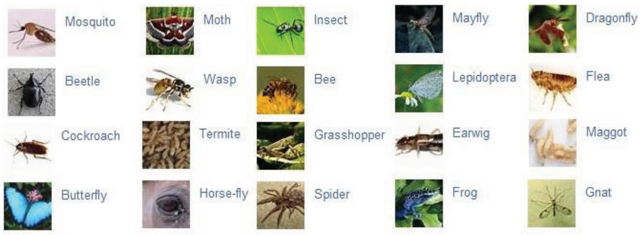


Figure 4: [Best viewed in color] *Bing Images* results (“`fly`”).

Note that the aforementioned limitations are not only confined to social images search engines. Even for query-specific image categorization techniques provided by Web image search engines (*e.g., Google Images* (`images.google.com`), *Bing Images* (`www.bing.com/images`)), where data associated with images are not as sparse as social images, there is little evidence whether they maximally cover the results. For example, consider the image categories generated by *Google Images* (Figure 3) and *Bing Images* (Figure 4) for the query “`fly`”[3]. Despite having significantly larger datasets

[1] We assume that the tags are high-level semantic concepts assigned by image uploaders or annotators.

[2] In the sequel, we use *tag* and *concept* interchangeably.

[3] The results are last accessed on August 13, 2013.

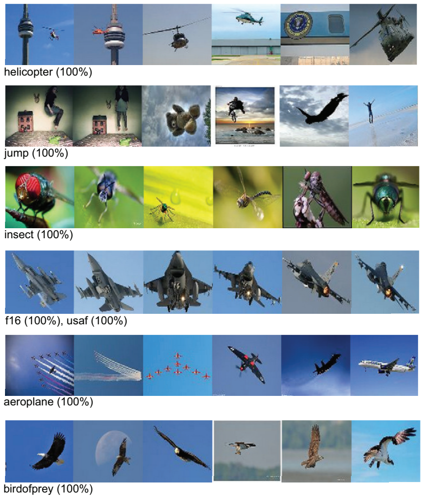


Figure 5: [Best viewed in color] Concept-preserving image clusters generated by PRISM for the query "fly".

and richer set of web text annotations, these search engines still construct relatively limited variety of concepts. The concepts suggested by them are mostly restricted to insects. Clearly, they have missed out other fly-related concepts such as the act of jumping, planes, helicopter, and birds.

In this paper, we propose a novel query-specific social image search results summarization algorithm called PRISM[4] (concept-**PR**eserving social **I**mage **S**earch su**M**marization) that constructs high quality summary of image search results based on concept-preserving and visually coherent clusters which maximally cover the result set. Figures 2(a) and 5 depict subsets of clusters constructed by PRISM for the queries "fruit" and "fly", respectively. Each cluster is represented by *minimal* tag(s) shared by *all* images in it. Due to the concept-preserving nature, the images in a cluster form an equivalence class with respect to the tags. Consequently, any image in each cluster can be selected as an exemplar. For instance, any image in the "pear" cluster can be chosen as an exemplar to represent it (*e.g.,* first three images are chosen in this example). Also observe that in contrast to *Google Images* and *Bing Images*, PRISM generates clusters representing wider variety of concepts related to "fly" (Figure 5) that maximally cover the result set.

Any query-specific image search results summarization presents several non-trivial challenges. The set of images to be summarized is not predetermined. Hence, the summarization method cannot preprocess the underlying images *apriori*. Additionally, simply leveraging traditional image clustering techniques may not generate high-quality summary due to the requirement that any summary must be concept-preserving and cover as many images as possible in the result set. Furthermore, it has to be robust to a wide variety of queries and result sizes. To address these challenges, PRISM explores the concept space (*i.e.,* tag space) to seek for visually coherent cluster of images. Note that a single-dimensional exploration of the concept space, however, may not yield visually related images. As such, PRISM models the exploration of the visual-concept space using a graph model. Specifically, it first constructs a *visual similarity graph* G where the nodes are images in the search results and the edges represent visual similarities between pairs of images. Then it *optimally* decompose G into a set of *concept-preserving subgraphs* based on some *summarization objectives* that encompass the aforementioned features of image clusters. Particularly, images in each subgraph represents a concept-preserving cluster. Following that, PRISM performs a series of image set *compression* to simplify the subgraphs to form the final set of concept-preserving subgraphs. Lastly, one or more exemplar images from each subgraph is selected to form the *exemplar summary* as depicted in Figure 2(a).

The rest of the paper is organized as follows. Section 2 reviews related research and Section 3 defines the research problem. Section 4 presents the PRISM algorithm. We investigate the performance of PRISM in Section 5. Section 6 concludes the paper. A preliminary two-page poster of PRISM is presented in [14].

2. RELATED WORK

Exemplar-based Summarization. One approach of image summarization is to find a set of exemplars that summarize the image set (*e.g., Bing Images*). Raguram and Lazebnik [12] propose a method that constructs a joint clustering on image descriptors and tag topic vectors independently before obtaining their intersection. Following that, a quality ranking learned from labeled images is used to select iconic images. In [7], a set of exemplars is identified using a sparse *Affinity Propagation* (AP) approach. Simon *et al.* [15] formulate the scene summarization problem for selecting a representative set of images of a given scene. A k-means-based greedy method is proposed to compute clusters using visual features. The mostly likely tag associated with each cluster is then determined using a probabilistic measure. Xu *et al.* [20] evaluates visual and textual information jointly using a technique known as *homogeneous* and *heterogeneous message propagation* to identify exemplar images. The method extends the AP algorithm to support heterogeneous messages from visual and textual feature spaces. In contrast to PRISM, these approaches do not attempt to ensure that all other images can be properly clustered by their exemplars (and their tags) in a concept-preserving manner. Additionally, they do not ensure that the exemplars maximally cover the image set.

Clustering-based Summarization. Clustering an image collection to find blocks of similar images is another approach to address the summarization problem. Several methods cluster images purely based on the semantic concepts associated with the images, such as tags [17, 19]. These methods, however, cannot assess and guarantee the visual coherence of the clustered images. Other methods consider only the visual similarity among images [8].

Clustering of tagged social images by considering both visual and textual features can be viewed as multi-modal clustering consisting of two types, namely *early fusion* and *late fusion*. In *early fusion*, the modalities are combined and evaluated simultaneously. Cai et al. [2] exploit a combination of visual, textual, and edge information of Web images to construct a *relationship* graph. Spectral clustering is then applied to obtain clusters of related images. No attempt, however, is made to associate a concept with each cluster. Instead, surrounding texts around the cluster of images are used to index the images. Heterogeneous clustering of visual, textual, and edge data is also studied by Li *et al.* [9]. Blaschko and Lampert [1] introduce a correlational spectral clustering approach on images with associated text. The technique is based on kernel canonical correlation analysis that finds projections of the image and text data. Rege *et al.* [13] propose a tripartite graph partitioning framework on clustering Web images and text. The framework obtains partitions of correlated web images, textual information, and visual features. *Late fusion* computes the clustering on each modality independently. These clusterings are then integrated to form the final multi-modal clustering. Moëllic *et al.* [10] propose a clustering method based on shared nearest neighbors. Unlike PRISM which

[4] A prism can be used to break a beam of light up into its constituent spectral colors (the colors of the rainbow). Similarly, the PRISM algorithm breaks the result image set into distinct image clusters.

considers the modalities in tandem, it clusters images in a sequential manner–first based on image tags, then on visual descriptors.

Generalized multi-modal clustering methods in most cases do not associate each cluster with a tag concept for user interpretation and visualization. As such, one has to associate tag(s) to each image cluster as a post-processing step. As remarked earlier, such tag-image cluster associations rarely preserve concepts as opposed to the tight tag-cluster integration attained by PRISM where all images in a cluster share the same concept(s). Lastly, unlike PRISM these techniques do not seek to find a concise set of images that can maximally cover the entire result set.

3. PROBLEM FORMULATION

3.1 Terminology

Given a search query $Q = \{q_1, q_2, \ldots, q_c\}$ consisting of one or more keywords (tags), suppose that a social image search engine (*e.g., Flickr*) retrieves a list of result images $\mathcal{D}$ satisfying Q. By abusing the notation of lists, let $\mathcal{D} = \{i_1, i_2, \ldots, i_n\}$ and $|\mathcal{D}| = n$. Each image $i \in \mathcal{D}$ comprises of: (a) a d-dimensional visual feature vector representing visual content of the image; and (b) a set of tags $T_i = \{t_1, t_2, \ldots, t_{|T_i|}\}$ associated with i. Note that $Q \subseteq T_i$.

The visual similarities among images in $\mathcal{D}$ is represented as a *visual similarity graph* $G = (V, E, w)$, where V is the set of images in $\mathcal{D}$ and E is a set of undirected edges between visually similar images. The function $w : E \rightarrow \mathbb{R}$ assigns *weight* to each edge to indicate the degree of visual similarity between images. Figure 6(a)(i) illustrates a visual similarity graph.

Given a set of tags T, a *concept-preserving subgraph* (*concept subgraph* for brevity), denoted by $C_T = (V_T, E_T, T)$, is a subgraph of G induced by $V_T \subseteq V$. Every image in the subgraph shares the set of tags T, *i.e.,* $T \subseteq T_i \ \forall \ i \in V_T$. We use concept subgraphs to model a set of images that preserves a set of concepts represented by T. In Figure 6(a)(i), the subgraph induced by the node set $\{v1, v2, v3\}$ is an example of a concept-preserving subgraph where $T = \{$"surf"$\}$. Every image in the subgraph shares all concepts in T.

A concept subgraph in G can be concisely represented by an *exemplar node* labeled with T. Figure 6(a)(ii) depicts a set of exemplar nodes (represented by dashed circles) with labels "surf", "beach", "sea", and "sun". These nodes represent the concept subgraphs induced by $\{v1, v2, v3\}$, $\{v8, v9, v10\}$, $\{v4, v5, v6, v7, v9\}$, and $\{v11, v12, v13, v14\}$, respectively.

3.2 Search Results Summarization Problem

We now formally define the problem of social image search results summarization. Intuitively, it can be formulated as the *optimal* decomposition of a visual similarity graph G into a set of concept subgraphs from which exemplar images are drawn to create the summary. Let us elaborate on it with an example. Consider the subgraph in Figure 6(a)(i) induced by the node set $\{v3, v4, v6, v9\}$ sharing no common concept and the concept subgraph induced by $\{v1, v2, v3\}$ sharing the concept $T = \{$"surf"$\}$. Notice that any image represented by the exemplar node of $\{v1, v2, v3\}$ (Figure 6(a)(ii)) can be selected as an exemplar summary for the "surf" images (Figure 6(a)(iii)). However, with no shared concepts in the node set $\{v3, v4, v6, v9\}$, it is less obvious how the entire subgraph can be represented with an exemplar image. Hence, if one can *optimally* decompose G into concept subgraphs, then one can meaningfully represent G with a concise set of exemplar nodes from which the exemplar summary of the result set can be generated. This is the key intuition behind summarization of G using concept subgraphs.

More specifically, a *decomposition* of G generates a set of concept subgraphs $\mathcal{S} = \{C_{T^1}, C_{T^2}, \ldots C_{T^k}\}$ and a *remainder* subgraph R, such that the image set in G is union of all images in $\mathcal{S}$ and R. Each $C_{T^i} \in \mathcal{S}$ can be represented by an exemplar node; the remainder subgraph R represents the region of G not covered by $\mathcal{S}$ (*i.e., R* is the subgraph induced by the set $V \setminus \bigcup_{C_T \in \mathcal{S}} V_T$). For example, the visual similarity graph in Figure 6(a)(i) is decomposed into $\{C_{surf}, C_{beach}, C_{sea}, C_{sun}\}$ and R where C_{surf}, C_{beach}, C_{sea}, and C_{sun} are represented by exemplar nodes "surf", "beach", "sea", and "sun", respectively, and $R = \{v15, v16\}$. Our decomposition allows overlap among subgraphs in $\mathcal{S}$ (*e.g.,* overlap between C_{beach} and C_{sea}).

A keen reader may observe that there are numerous ways of decomposing G into $\mathcal{S}$ and R. However, not all decompositions result in high quality summary. For instance, suppose we decompose G into concept subgraphs $\{v1, v8, v11\}$ and $\{v1, v14\}$ represented by exemplar nodes "nikon" and "boat", respectively. Clearly, this decomposition poorly summarizes G because the images within each subgraph have low visual similarities (*e.g.,* subgraph $\{v1, v8, v11\}$ contains no edges) and only 4 out of 16 images are represented by exemplar nodes. Hence, it is important to *optimally* decompose G so that it can facilitate high quality summary construction.

Let $\mathcal{E}$ be the family of all concept subgraphs of G representing all *potential* concept-preserving clusters. Obviously, $\mathcal{E}$ can easily comprise of prohibitively large number of overlapping subgraphs; rendering it impractical for summary construction. It is therefore pertinent to identify a small subset of $\mathcal{E}$ that is sufficient to represent and summarize G. Hence, we want to find a subset $\mathcal{S} \subset \mathcal{E}$ that optimally decomposes G based on some *summarization objectives* from which a concise summary can be generated. Specifically, a *summary* of G is the set of exemplars obtained from $\mathcal{S}$ by mapping every concept subgraph $C_T \in \mathcal{S}$ to its associated exemplar(s). In Figure 6(a), the summary of the visual similarity graph is the set of exemplars that represents the concepts "surf", "beach", "sea", and "sun". The remainder subgraph $R = \{v15, v16\}$ represents images "missed" by the summary. We consider the following *summarization objectives* for optimal decomposition of G:

- *Visual coherence.* The *visual coherence* of $\mathcal{S}$ is defined as:

$$coherence(\mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{C_T \in \mathcal{S}} \frac{\sum_{e \in E_T} w(e)}{|E_T|} \tag{1}$$

  The $coherence(\mathcal{S})$ value reflects the average weight of visually similar images in each $C_T \in \mathcal{S}$. Higher visual coherence means the images are more visually similar to each other.

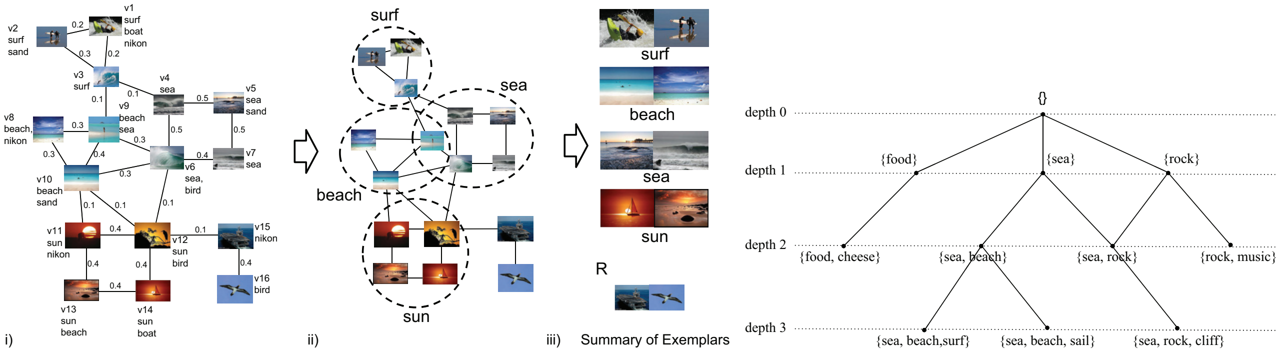
- *Distinctiveness.* Intuitively, a pair of exemplar nodes that represent two disjoint subgraphs is more informative that a pair that represent identical subgraphs. Thus, a decomposition that creates clean separation of concept subgraphs is desirable. We quantify this objective with the *distinctiveness* measure. It measures concept subgraph redundancies, such that the greater the redundancies, the lower the distinctiveness value. Formally, *distinctiveness* of $\mathcal{S}$ is defined as:

$$distinctiveness(\mathcal{S}) = \frac{\left|\bigcup_{C_T \in \mathcal{S}} V_T\right|}{\sum_{C_T \in \mathcal{S}} |V_T|} \tag{2}$$

- *Coverage.* A set of concept subgraphs $\mathcal{S}$ that well represents G is preferable. We use the notion of *coverage* to measure this. Intuitively, it quantifies how many images from the image set V appears in $\mathcal{S}$. Formally, it is defined as:

$$coverage(\mathcal{S}) = \frac{\left|\bigcup_{C_T \in \mathcal{S}} V_T\right|}{|V|} \tag{3}$$

  Note that $coverage(\mathcal{S})$ is 1 if all images in V are selected in $\mathcal{S}$. As we shall see later, there is a trade-off between maximizing coverage or distinctiveness.



(a) Summarization process.

(b) Refinement.

Figure 6: [Best viewed in color] Illustration of the social image search results summarization in PRISM.

DEFINITION 1. *Let Q be a search query on a social image database and $\mathcal{D}$ be the set of search results. Given the visual similarity graph G of $\mathcal{D}$, the goal of the* **social image search results summarization problem** *is to find an optimal set of concept subgraphs $\mathcal{S}$ such that coherence($\mathcal{S}$), coverage($\mathcal{S}$) and distinctiveness($\mathcal{S}$) are maximized. Following that, the* ***exemplar summary*** *$\mathcal{M}$ is constructed by selecting from each concept subgraph $C_T \in \mathcal{S}$ an exemplar set (comprising of $1 \leq m \leq 3$ images in C_T) and its associated concept.*

Let us illustrate the problem definition with an example. Consider Figure 6(a) and two sets of concept subgraphs $\mathcal{S}_1 = \{\{v1, v2, v3\}, \{v8, v9, v10\}, \{v4, v5, v6, v7\}, \{v11, v12, v13, v14\}\}$ and $\mathcal{S}_2 = \{\{v1, v8, v11\}, \{v1, v14\}\}$. Observe that $\mathcal{S}_2$ has lower distinctiveness (while every image belongs to at most one concept in $\mathcal{S}_1$, several images belong to two concepts in $\mathcal{S}_2$). The clusters of images in $\mathcal{S}_2$ also have lower visual coherence (fewer edges within subgraphs). The coverage of $\mathcal{S}_2$ is also lower than $\mathcal{S}_1$. Hence, $\mathcal{S}_1$ is superior to $\mathcal{S}_2$.

To solve the problem in Definition 1, we propose a weighted minimum k-set cover optimization model [5]. It includes a cost model that incurs a weight (*i.e.,* cost) every time a subgraph is added as concept subgraph or as remainder subgraph. For each concept subgraph, we incur a *visual incoherence cost*, the inverse of visual coherence of a concept subgraph, for choosing visually incoherent images (maximize *coherence*($\mathcal{S}$)). For each remainder subgraph, we incur a *remainder penalty* cost for choosing large remainder subgraphs (maximize *coverage*($\mathcal{S}$)). Given the cost model, we find the minimum weight (cost) of subgraphs needed to cover V, penalizing redundant subgraphs that add little to the summary since every subgraph added incurs a cost (controlling *distinctiveness*($\mathcal{S}$)).

DEFINITION 2. *Given the visual similarity graph G of $\mathcal{D}$, let $\mathcal{E}$ be the family of all concept subgraphs of G and $\mathcal{F}$ be the family of all subgraphs of G. Let k be the cardinality constraint. The optimal $\mathcal{S} \cup \mathcal{R}$, where $\mathcal{S} \subset \mathcal{E}$ (set of concept subgraphs) and $\mathcal{R} \subset \mathcal{F}$ (set of remainder subgraphs), is the minimum cost set that covers V:*

$$\underset{\mathcal{S} \cup \mathcal{R}}{\arg\min}\ f(\mathcal{S} \cup \mathcal{R}) = \underset{\mathcal{S} \cup \mathcal{R}}{\arg\min} \sum_{C_T \in \mathcal{S}} c(C_T) + \sum_{R \in \mathcal{R}} c(R) \qquad (4)$$

subject to $V = \bigcup_{C_T \in \mathcal{S}} V_T \cup \bigcup_{V_R \in \mathcal{R}} V_R$ and $|\mathcal{S}| + |\mathcal{R}| \leq k$, where the visual incoherence *cost function $c : \mathcal{E} \rightarrow \mathbb{R}$ and the* remainder penalty *cost function $c : \mathcal{F} \rightarrow \mathbb{R}$ are defined as follows:*

$$c(C_T) = \frac{|E_T|}{\sum_{e \in E_T} w(e)} \qquad c(R) = (|V_R| + 1) \max_{C_T \in \mathcal{E}} c(C_T)$$

Observe that that we model the scenario whereby having images in a remainder subgraph will always incur higher penalty than representing them with a concept subgraph (even if visual coherence is low). We now prove that an optimal solution of the problem is a set of concept subgraphs $\mathcal{S}$ and at most a single remainder subgraph R, and the remainder subgraph does not overlap with $\mathcal{S}$.

THEOREM 1. *If the solution $\mathcal{S}_0 \cup \mathcal{R}_0$ of the social image search results summarization problem is optimal, then $|\mathcal{R}_0| \leq 1$.*

PROOF. Assume by contradiction that $|\mathcal{R}_0| > 1$. $\mathcal{R}_0$ covers the set $\bigcup_{R \in \mathcal{R}_0} V_R$. The cost incurred by sets in $\mathcal{R}_0$ is $|\mathcal{R}_0| \max_{C_T \in \mathcal{E}} c(C_T)$ $+(\max_{C_T \in \mathcal{E}} c(C_T)) \sum_{R \in \mathcal{R}_0} |V_R|$. We show that we can replace $\mathcal{R}_0$ with a single remainder subgraph and incur a lower cost. Let $\mathcal{R}' = \{\bigcup_{R \in \mathcal{R}_0} V_R\}$. The singleton $\mathcal{R}'$ covers the same set of vertices with lower cost and lower set cover cardinality. □

THEOREM 2. *If $\mathcal{S}_0 \cup \mathcal{R}_0$ is optimal, then the following holds: $\bigcup_{C_T \in \mathcal{S}_0} V_T \cap \bigcup_{R \in \mathcal{R}_0} V_R = \emptyset$.*

PROOF. Assume by contradiction that $\bigcup_{C_T \in \mathcal{S}_0} V_T \cap \bigcup_{R \in \mathcal{R}_0} V_R \neq \emptyset$. Let $\mathcal{R}' = \{\bigcup_{R \in \mathcal{R}_0} V_R \setminus \bigcup_{C_T \in \mathcal{S}_0} V_T\}$. $\mathcal{S}_0 \cup \mathcal{R}'$ covers the same set of vertices with lower cost incurred. □

Since weighted k-set cover problem is NP-hard [5], in the next section we present a greedy algorithm to address it.

4. THE PRISM ALGORITHM

An algorithm that solves the aforementioned summarization problem must resolve two key issues: (a) a structure to allow efficient enumeration of concept subgraphs in $\mathcal{E}$ and (b) a method to efficiently find an optimal subset of $\mathcal{E}$ and $\mathcal{F}$ that maximizes the summarization objectives. The PRISM algorithm (Algorithm 1) is designed to achieve these. It consists of five key phases: the *visual similarity graph construction* phase (Line 1), the *$\mathcal{E}$-construction* phase (Line 2), the *decomposition* phase (Line 3), the *summary compression* phase (Line 4), and the *exemplar summary generation* phase (Lines 5-9). Given a search results $\mathcal{D}$, a visual similarity graph G is first constructed. The *$\mathcal{E}$-construction* phase then constructs the family of concept subgraphs of G. Subsequently, the *decomposition phase* performs a combinatorial optimization to decompose G into a set of concept subgraphs $\mathcal{S} \cup \mathcal{R}$ based on the three summarization objectives. Images in each subgraph represent a concept-preserving cluster (Recall from Section 1). Note that state-of-the-art graph clustering techniques [1,3,13] cannot be directly leveraged to identify these clusters as they do not preserve concepts, typically generate non-overlapping clusters, and do not maximally cover the entire graph. The *summary compression* process "compresses" $\mathcal{S}$ to form a summary at reduced level of detail (denoted by $\mathcal{V}$). The final phase involves selection of one to three exemplar images from each concept subgraph in $\mathcal{V}$ to form $\mathcal{M}$. Since the last phase is straightforward, we now proceed to elaborate on the first four phases.

Algorithm 1: The PRISM algorithm

```
Input: User query $\mathcal{Q}$, Set of images $\mathcal{D}$, $\delta$, $k$.
Output: Exemplar summary $\mathcal{M}$.
$G \leftarrow ConstructVisSimGraph(\mathcal{D}, \delta)$;
$\mathcal{E}^* \leftarrow \mathcal{E} - Constructor(G)$;
$\mathcal{S} \leftarrow Decompose(\mathcal{E}^*, k)$;
$\mathcal{V} \leftarrow Compress(\mathcal{S})$;
$\mathcal{M} \leftarrow \emptyset$;
forall the $C_T = (V_T, E_T, w, T) \in \mathcal{V}$ do
    select $m$ images in $V_T$ as exemplar;
    associate $m$ images with tag $T$;
    $\mathcal{M} \leftarrow \mathcal{M} \cup (m, T)$;
return $\mathcal{M}$;
```

Algorithm 2: The $\mathcal{E} - Constructor$ algorithm.

```
Input: Visual similarity graph $G$.
Output: A set of concept-preserving subgraphs $\mathcal{E}^*$.
$(V, E, w) \leftarrow G$;
$i \leftarrow 0$;
$V_i \leftarrow \{C_T^0 = (V, E, \emptyset)\}$;
$\mathcal{E}^* \leftarrow \emptyset$;
while $V_i$ is not empty do
    $V_{i+1} \leftarrow \emptyset$;
    forall the $C_T \in V_i$ do
        $R_{i+1} \leftarrow$ refinements of $C_T$;
        $V_{i+1} \leftarrow V_{i+1} \cup R_{i+1}$ ;
    $\mathcal{E}^* \leftarrow \mathcal{E}^* \cup V_{i+1}$;
    $i \leftarrow i + 1$;
return $\mathcal{E}^*$;
```

4.1 Visual Similarity Graph Construction Phase

Since the visual similarity graph G is query-dependent, it needs to be constructed on-the-fly. To this end, we adopt cosine similarity to measure the visual similarity between any two images as follows: $Sim = L^{-1/2}A^TAL^{-1/2}$ where A is the $n \times d$ matrix of image set visual features, A^TA encodes the inner-product of the image feature vectors, and $L^{-1/2}$ is a $n \times n$ diagonal matrix that encodes normalization of each feature vector. Given the similarity matrix, we construct the visual similarity graph G as follows. Let V be the set of images. We add an edge in E between two images i and j if $Sim_{ij} > \delta$. The weight of this edge is Sim_{ij} and the *edge density parameter* δ is user-defined, controlling the edge density of G.

4.2 $\mathcal{E}$-Construction Phase

Recall that it is unrealistic to exhaustively explore $\mathcal{E}$. Hence, we propose a method that explores $\mathcal{E}$ *selectively* using a directed acyclic graph (DAG) *exploration model* (DAG *model* for brevity). The main objective is to provide an exploration structure for enumerating concept subgraphs. We denote this exploration by $\mathcal{E}^*$.

We first outline the construction of the DAG model. With exception of the root node, every node in the DAG represents a concept subgraph. Let C_T^0 be the root node of the DAG at depth $d = 0$, where $C_T^0 = (V, E, \emptyset)$ represents G with no shared concepts (*i.e.,* not a concept subgraph). Given C_T^d, we construct C_T^{d+1} as follows. For each C_T^i in C_T^d, a *refinement* of C_T^i is a concept subgraph $C_T^{i+1} = (V_T^{i+1}, E_T^{i+1}, T^{i+1})$ that satisfies the following:

1. T^{i+1} is T^i and one additional concept t', *i.e.,* $T^{i+1} = T^i \cup t'$
2. V_T^{i+1} is the set of all images in V_T^i sharing T^{i+1} and $V_T^{i+1} \neq V_T^i$
3. C_T^{i+1} induced by V_T^{i+1} has at least one edge (at least a pair of images are visually similar)

For example, consider the DAG model in Figure 6(b) where each node represents a concept subgraph (labeled with the shared concept T for brevity). The $\{sea, beach\}$ node is a refinement of $\{sea\}$. Similarly, $\{sea, beach, surf\}$ node is a refinement of $\{sea, beach\}$. Observe that a refinement C_T^{i+1} represents images that share one more concept than in C_T^i (by first criteria). In fact, each subgraph at depth d contain images that share d concepts. Also, it is always a proper subgraph of C_T^i so that there are no redundant subgraphs (by second criteria). We ignore any C_T^{i+1} without an edge because it has no potential to form visually coherent images (by third criteria).

Intuitively, the refinements as subgraphs of C_T^i represent finer-grained concepts. At each depth d, we construct finer-grained refinements of its parent graphs. Hence, starting with C_T^0, we build a hierarchy of refinements to form the DAG model. We recursively identify the next set of refinements of the DAG at $d = 1, 2, 3, \ldots$ in similar way until $V^d = \emptyset$. Algorithm 2 outlines the construction of $\mathcal{E}^*$. Let the maximum depth of the DAG be d. Then the worst case time complexity is $O(\sum_i^d \binom{m}{i})$ where $m = |\cup_{i \in V} T_i|$ and at any depth $i > 0$, one can construct up to $\binom{m}{i}$ concept subgraphs. Note that despite its exponential complexity, as we shall see later, in practice this phase completes quickly as users are typically interested in summary of the top-n (*e.g.,* $n < 2000$) results instead of the entire result set.

4.3 Decomposition Phase

In this phase, we find a subset $\mathcal{S} \subset \mathcal{E}^*$ and $\mathcal{R} \subset \mathcal{F}$ that optimally decomposes G. Recall from Definition 2 our goal of finding the subset $\mathcal{S} \cup \mathcal{R} \subset \mathcal{E}^* \cup \mathcal{F}$ that minimizes $\sum_{C_T \in \mathcal{S}} c(C_T) + \sum_{R \in \mathcal{R}} c(R)$ subject to vertex cover and cardinality constraints (Equation 4). Due to its computational hardness, we adopt a H_k-approximation greedy algorithm, where $H_k = \sum_{i=1}^k \frac{1}{i}$ [5]. Algorithm 3 outlines the greedy strategy of selecting $\mathcal{S} \cup \mathcal{R}$ to minimize $\sum_{C_T \in \mathcal{S}} c(C_T) + \sum_{R \in \mathcal{R}} c(R)$. The basic idea is to select, at each iteration, $X \in \mathcal{E}^* \cup \mathcal{F}$ (X is either a concept subgraph or a remainder subgraph) so that X has the lowest $c(X)/n$ cost incurred, where n is the number of new vertices covered by X (Lines 5-11). Intuitively, we pay $c(X)$ to cover an extra n vertices, and the subgraph with lowest $c(X)/n$ contributes maximum value by having the lowest cost per vertex coverage gained.

Recall from Theorems 1 and 2 that there should be at most one remainder subgraph that is disjoint with $\mathcal{S}$. Since $c(X)/n$ of a remainder subgraph is always larger than $c(X)/n$ of a concept subgraph, the greedy algorithm will always add concept subgraphs before remainder subgraphs, as long as there is gain in coverage. Therefore, C_T is always added until k concept subgraphs have been selected. Then R is the final remainder subgraph induced by the unselected images, which incurs a cost $c(R)$. Notice that each iteration involves a single pass through the subgraphs in $\mathcal{E}^*$. With a total of k iterations, the algorithm involves processing $k|\mathcal{E}^*|$ subgraphs, which in the worst case evaluates $O(k|\mathcal{E}^*||V|)$ images.

4.4 Summary Compression Phase

The preceding phase finds an optimal collection of concept-preserving clusters *without* constraining each cluster size. This is beneficial as it enables us to select the "best" combination of clusters with highest visual coherence. On the other hand, there is a lack of control over the *summary granularity* if each concept subgraph in the constructed $\mathcal{S}$ is used for creating the exemplar summary as $\mathcal{S}$ may contain too finely-grained clusters for presentation to users. We assume that a user expects a summary at a particular summary granularity. For instance, if a user wants a broad overview of the search result, then a summary of 5 exemplars may be preferable to a summary of 50 exemplars. On the other hand, if a user prefers a detailed summary, then the summary with 50 exemplars is better.

At first glance it may seem that one may adjust the parameter k to achieve the desired summary granularity. However, as we shall see later, k significantly affects the coverage and distinctiveness of the summary. Hence an alternative approach that can modify the summary granularity without affecting coverage and distinctiveness is desirable. In this phase, we address this issue by building multiple summaries at varying summary granularity by *aggregating* concept subgraphs. Specifically, a *compressed concept subgraph set* $\mathcal{S}^i$ is formed by aggregating concept subgraphs in $\mathcal{S}$ to form another set of subgraphs of lower *summary granularity*. For example, assume that $\mathcal{S}$ contains two subgraphs with $T^1 = \{boat, sail, rock\}$ and $T^2 = \{rock, cliff\}$. Then these two subgraphs can be *aggregated* into a larger subgraph sharing the $\{rock\}$ concept. Consequently, it compresses two concept subgraphs into a single subgraph.

We introduce a *multilevel compression* scheme that aggregates concept subgraphs iteratively. Given the initial $\mathcal{S}$, we construct a list of concept subgraph set with increasingly smaller size. Formally, we construct a list $[\mathcal{S}, \mathcal{S}^1, \mathcal{S}^2, \ldots, \mathcal{S}^d]$ such that $\forall i, j$, $|\mathcal{S}^i| > |\mathcal{S}^j|$ if $i < j$. We call each $\mathcal{S}^i$ a *compressed concept subgraph set* of $\mathcal{S}$. Each successive set $\mathcal{S}^{i+1}$ is a compressed representation of its predecessors. Observe that if a user wants a detailed summary of the search result, then $\mathcal{S}$ is most appropriate for generating exemplar summaries. If a broader overview is desired, then a compressed set provides more concise view of the result set. In PRISM, by default we use $\mathcal{S}^d$ to create the exemplar summary. If desired, the user may drill into more detailed summaries.

We now elaborate on the construction of $\mathcal{S}^{i+1}$ from $\mathcal{S}^i$. Given $\mathcal{S}^i$, the successor $\mathcal{S}^{i+1}$ is constructed by *contracting* pairs of concept subgraphs. The contraction of pairs C_{T^1} and C_{T^2} removes both subgraphs from the set and replaces them with $C_{T^1 \cup T^2} = (V_{T^1} \cup V_{T^2}, E_{T^1} \cup E_{T^2})$. Figures 7(a)-(b) illustrate the contraction of two concept subgraphs with $T^1 = \{sea, surf, hawaii\}$ and $T^2 = \{sea, surf, nikon\}$ into a subgraph with $T^3 = \{sea, surf\}$. Through successive subgraph pair contractions, we obtain increasingly compressed concept subgraph set.

How do we determine which pairs of concept subgraphs in $\mathcal{S}$ to contract? Intuitively, one prefers to contract conceptually similar subgraphs while keeping conceptually distinct subgraphs uncontracted. Given $C_{T^1} \in \mathcal{S}^i$ and $C_{T^2} \in \mathcal{S}^i$, we say that C_{T^1} and C_{T^2} is *coupled* if all images in $V_{T^1} \cup V_{T^2}$ share a non-empty set of concepts (*i.e.,* all images have at least one common concept). Only coupled concept subgraphs can be contracted; if not, $\mathcal{S}^{i+1}$ may contain subgraphs that violate the concept preservation property of concept subgraphs. We can represent these couplings in $\mathcal{S}^i$ using a *coupling graph*. A *coupling graph* of $\mathcal{S}^i$ is a graph $G_c^i = (\mathcal{S}^i, E_c^i)$ where each node is a concept subgraph. We add an edge in G_c between $C_{T^1} \in \mathcal{S}^i$ and $C_{T^2} \in \mathcal{S}^i$ *iff* C_{T^1} and C_{T^2} is coupled (thus valid candidate for contraction). Each edge is weighted, indicating the degree of coupling between the coupled subgraphs. Here ω_{12} is the *coupling weight* of the edge between C_{T^1} and C_{T^2} and is defined as: $\omega_{12} = \sum_{t \in T} OR(Q, t)$ where T is the set of concepts shared among all images in $V_{T^1} \cup V_{T^2}$ and $OR(t, Q)$ is the *relevance* of a tag t to query Q using odds ratio:

$$OR(t, Q) = \max_{q \in Q} \frac{Pr(q, t)Pr(q^c, t^c)}{Pr(q^c, t)Pr(q, t^c)} \qquad (5)$$

In the above equation, $Pr(x, y)$ is the probability of co-occurrence of events x and y and x^c denotes the event of x *not* occurring. We utilize the co-frequency of the relevant tags to determine the probability values. Given tags q and t, let I_q and I_t be the sets of images having tags q and t, respectively. The co-frequency between q and t is simply $|I_q \cap I_t|$ and $Pr(q, t) = |I_q \cap I_t|/|V|$. Observe that the coupling weight depends on concept relevance to the user query as well as number of shared concepts. Figure 7(a) is an example of a coupling graph.

Algorithm 3: The *Decompose* algorithm.

```
Input: $\mathcal{E}^*$, $k$.
Output: A set of concept-preserving subgraph $\mathcal{S}$
$\mathcal{S} \leftarrow \emptyset$;
repeat
    $mincost \leftarrow \infty$;
    $bestcluster \leftarrow \emptyset$;
    forall the $C_T \in \mathcal{E}^* \setminus \mathcal{S}$ do
        $n \leftarrow |V_T \setminus \bigcup_{C \in \mathcal{S}} V|$;
        $f \leftarrow c(C_T)/n$;
        if $f < mincost$ and $n > 0$ then
            $mincost \leftarrow f$;
            $bestcluster \leftarrow \{C_T\}$;
    $\mathcal{S} \leftarrow \mathcal{S} \cup bestcluster$;
until $|\mathcal{S}| > k$;
return $\mathcal{S}$;
```

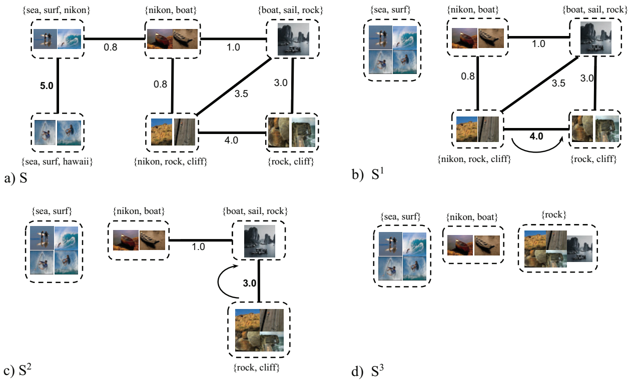


Figure 7: Summary compression phase. For clarity, we depict a node representing a concept subgraph by its images only.

Algorithm 4 outlines the summary compression phase. We describe it using the example in Figure 7(a). To select pairs of subgraphs for contraction, we employ the following contraction scheme. (a) For each $\mathcal{S}^i$, choose the highest weighted edge in the coupling graph and contract the nodes of this edge (Lines 5-12). This results in compression of $\mathcal{S}^i$ to $\mathcal{S}^{i+1}$ (Lines 13-15). (b) Repeat the process for the next $\mathcal{S}^i$ until its coupling graph has no edges (Lines 4-16). Figure 7 shows an example of this scheme. Notice that each iteration evaluates the pairwise concept subgraphs in $|\mathcal{S}|$. Thus, every iteration evaluates $|\mathcal{S}|^2$ subgraphs. If we assume the worst case which merges all concept subgraphs until $|\mathcal{S}| = 1$, then this phase evaluates $O(|\mathcal{S}|^3)$ concept subgraphs.

5. EXPERIMENTS

PRISM is implemented in Java 1.7. In this section, we present the performance of PRISM. All experiments were executed on a Intel Core 2 Duo Linux machine with 4GB memory.

5.1 Experimental Setup

All experiments were conducted on the NUS-WIDE dataset [4] containing 269,648 *Flickr* images with visual features, tags and human-

Algorithm 4: The *Compress* Algorithm.

```
Input: Set of concept subgraphs $\mathcal{S}$
Output: Compressed set of concept subgraphs $\mathcal{S}_c$
$i \leftarrow 0$;
$\mathcal{S}^i \leftarrow \mathcal{S}$;
repeat
    $bestscore \leftarrow 0$;
    $bestpair \leftarrow \emptyset$;
    forall the $C_{T^1} \in \mathcal{S}^i$ do
        forall the $C_{T^2} \in \mathcal{S}^i$ st. $C_{T^1} \neq C_{T^2}$ do
            if $\omega(C_{T^1}, C_{T^2}) > bestscore$ then
                $bestscore \leftarrow \omega(C_{T^1}, C_{T^2})$;
                $bestpair \leftarrow \{C_{T^1}, C_{T^2}\}$;
    $\{C_{T^1}, C_{T^2}\} \leftarrow bestpair$;
    $\mathcal{S}^{i+1} \leftarrow \mathcal{S}^i \setminus \{C_{T^1}, C_{T^2}\}$;
    $\mathcal{S}^{i+1} \leftarrow \mathcal{S}^{i+1} \cup C_{T^1 \cup T^2}$;
    $i \leftarrow i + 1$;
until $bestscore = 0$;
$\mathcal{S}_c \leftarrow \mathcal{S}^{i+1}$;
return $\mathcal{S}$;
```

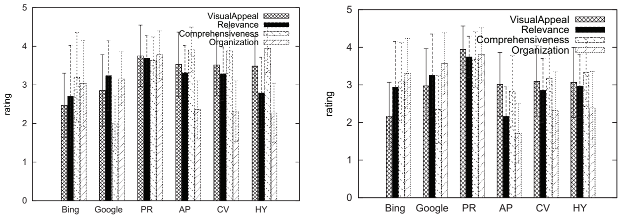
Table 1: Representative queries.

Type	Queries
Single-tag	asia (1.5), party (1.2), wedding (1.9), animals (1.4), art (1.3), city (1.5), rock (1.5), food (1.5), sun (1.4), sea (1.4), sky(1.7) nature (1.8), church (1.3), street (1.2), macro (1.7), bird (1.5),
Multi-tag	[sun, sea] (1.5), [sun, silhouette] (1.7), [blue, sea] (2.3) [street, art] (1.7), [sea, rock] (2.1), [blue, sky] (2.4), [rock, music] (2.2), [macro, insect] (2.7), [city, lights] (1.4), [flower, macro] (1.6), [cute, animals] (1.9), [red, food] (2.7), [graffiti, art] (2.3), [birthday, party] (1.2)

assigned labels in 81 categories. We use this dataset instead of original *Flickr* images due to the following reasons. First, the 81 human-assigned categories available in this dataset enable us to undertake quantitative evaluation of PRISM. Second, since users typically browse only top-*n* search results, it is reasonable to summarize only these results using PRISM. Consequently, the impact of dataset size on the summarization technique diminishes as the cost of retrieving these top-*n* results is orthogonal to PRISM.

As search results summarization is query-dependent, we selected 30 representative queries for our study. Since information related to most frequent queries on *Flickr* is not publicly available, we use a subset of frequent tags in *Flickr*[5] as a proxy for single-tag queries. Multi-tag queries are formed by adding tags to single-tag queries. Table 1 lists these queries (ignore for the time being the numeric values in parenthesis). For each query we selected up to 1000 top-ranked images ($|\mathcal{D}| = n = 1000$) from its search results to form its result set. Note that the query tag is ignored in the summarization process for all experiments to avoid bias due to the tag. All search results are obtained using a TagIR system following the best performing configuration in [16] on NUS-WIDE data collection.

Recall that the first step of PRISM is to construct a visual similarity graph. For this purpose, we used all 6 types of low-level visual features provided by NUS-WIDE dataset: 1) 64-D color histogram, 2) 144-D color correlogram, 3) 73-D edge direction histogram, 4) 128-D wavelet texture, 5) 225-D block-wise color moments, and 6) 500-D bag of words based on SIFT descriptions. Unless specified otherwise, we set $k = 150$ and $\delta = 0.05$.



(a) Single-tag queries (b) Multi-tag queries

Figure 8: User study.

Evaluation criteria. In addition to the coverage and distinctiveness measures outlined earlier, we introduce the *mean weighted global clustering coefficient* [11] to quantitatively measure the visual coherence of a summary $\mathcal{S}$. We define the *visual cohesiveness score* of a summary, denoted by VCS, as follows:

$$VCS(\mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{C_T \in \mathcal{S}} \frac{\sum_{T_\Delta} \sum_{e \in T_\Delta} w(e)}{\sum_T \sum_{e \in T} w(e)} \tag{6}$$

where w is the visual similarity weight function of C_T, the numerator $\sum_{T_\Delta} \sum_{e \in T_\Delta} w(e)$ sums over all closed triplets T_Δ in C_T, and $\sum_T \sum_{e \in T} w(e)$ sums over all triplets T in C_T [11].

To measure how well a concept is preserved in a cluster, we introduce the *concept preservation* metric. Given a summary $\mathcal{S}$, concept preservation of $\mathcal{S}$ is defined as:

$$ConceptPreservation(\mathcal{S}) = \frac{1}{|\mathcal{S}|} \sum_{C_T \in \mathcal{S}} \frac{\max_t |\{t : t \in T_i, i \in V_T\}|}{|V_T|} \tag{7}$$

where $ConceptPreservation(\mathcal{S}) \in (0, 1]$ and its value is 1 if for each cluster, every image shares at least a concept tag.

5.2 Comparison with the State-of-the-art

We compare PRISM (denoted by PR) with three representative summarization and clustering techniques: Canonical View Summarization (CV) [15], Affinity Propagation (AP) [6] and H^{2}MP (HY) [20]. AP and CV utilize only the visual features of the images in the summarization (or clustering) process. Tags are used in the post-processing to annotate the resultant clusters. The HY method utilizes both the visual and textual features of images. All tested methods share the same visual similarity matrix, and also share the same concept similarity matrix of tag co-occurrences. Where possible, the default parameters for each method were used. Otherwise, the parameters were adjusted to obtain reasonable results empirically and then remain fixed for multiple test sets. In addition, we qualitatively compare PRISM to visual summaries constructed by *Google Images* (Image Categories) and *Bing Images* (Related Topics).

User study. We first qualitatively evaluate the summarization results produced by the six approaches through a user study. We invited 12 unpaid volunteers (undergraduate and graduate students in computer science and business majors) to rate quality of the summaries. Nine of them had the experience of using image search engines. The remaining subjects are unfamiliar with image search. To avoid any bias on the evaluation, all the participants were selected such that they did not have any knowledge about the summarization technique deployed in PRISM[6]. Summaries generated by the algorithms are presented as a set of exemplars but without the names of the specific algorithms producing the summaries. For all methods, each exemplar is visually represented by three most relevant images and one or more concepts (*e.g.,* exemplars in Figure 2). For *Google Images* and *Bing Images*, the visual summary sections are

[5] Available at www.flickr.com/photos/tags/.

[6] None of the volunteers are authors of this paper.

Table 2: Separating power of the algorithms.

Algorithm	QC_2	QC_4	QC_6	QC_8
AP	0.854	0.594	0.478	0.412
CV	0.855	0.638	0.543	0.474
HY	0.867	0.627	0.520	0.450
PR	**0.955**	**0.956**	**0.911**	**0.930**

presented. Each participant was given one query at a time in random order (all 30 queries). They were allowed to take a break to refresh themselves if they feel tired during the evaluation process.

From the 30 queries in Table 1, a participant rates the quality of the summaries based on the following four questions.

QT_1: Is the summary visually appealing? (visual appeal)

QT_2: Are the exemplar summaries relevant to the query? (relevance)

QT_3: Is the summary comprehensive? (comprehensiveness)

QT_4: Is the summary well organized? Is it easy to understand at a glance? (organization)

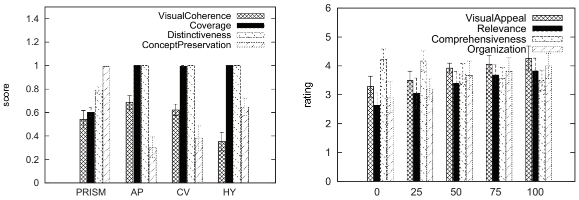
For each question, a participant rates the summary using a Likert scale, from 1 for most unsatisfactory to 5 for most satisfactory.

Figure 8(a) shows the results of the user study for single-tag queries. The rating for each question-algorithm pair is the average rating from multiple queries. The results clearly demonstrate the superiority of PRISM for QT_1-QT_4 (p-value in t-test is < 0.05 for each method) justifying the importance of concept preservation in order to obtain precise clusters. Figure 8(b) reports the results for multi-tag queries. We observe similar results for PRISM having the highest rating for visual appeal, relevance, and organization.

Notice that *Google*, *Bing* and PRISM summaries are perceived to be significantly better organized than other summarization approaches. We argue that this justifies the usefulness of having concept preserving summary with sparse tag exemplars. Figure 2 illustrates how exemplars with minimal tags are easier to interpret. We also observe that AP has the lowest relevance rating as it is likely to prioritize visually similar images over conceptually relevant images. Hybrid methods like PRISM and HY benefit from exploiting a richer set of heterogeneous data to guide the summarization process – visual features provide visual relationships between images, while textual/concept features provide semantic relationships. Notice the lower relevance rating for *Bing* compared to *Google* and PRISM. Upon closer inspection, we found that the relevance ratings for *Bing* summaries vary widely across different queries. Meanwhile, *Google* suffers from having lowest comprehensiveness because all query summaries only have *up to* five exemplars.

Separating power. Human evaluation is mostly limited by the scale of the evaluation. In this set of experiments, we evaluate the *separating power* of the algorithms using the NUS-WIDE dataset. More specifically, we combine the result sets of two or more queries to form a mixture set and then evaluate the effectiveness of a method in separating these images. To construct a mixture set of N queries (denoted by QC_N), an equal number of images are retrieved from N out of 81 ground-truth concepts. The ground-truth concepts selected are randomly determined, and for each test, we repeat with 10 random combinations of N concepts and obtain the mean score for the test. Every mixture set comprises 1000 images from the corresponding N query result sets (*i.e.,* each query in mixture set has $1000/N$ images). The combined images are then summarized using the summarization algorithms. We assume that a superior summary will partition the mixed images into their underlying query result sets with high accuracy.

As we are comparing clusterings in this evaluation, we perform the following post-processing for methods that construct only summaries. For HY, the Affinity Propagation-based method lends naturally to cluster construction by assigning each image to its exemplar. Likewise, for the CV approach, clusters are constructed by assigning each image to its closest canonical view. The label of a cluster in a summary is assigned based on majority voting. For instance, if a cluster contains 70% `insect` images and 30% `sports` images, then it assumes the `insect` class through majority vote, and the `sports` images are deemed mismatched.



(a) Evaluation metrics (b) Summary compression

Figure 9: Performance results.

Table 2 shows for each mixture set the fraction of images with matched assignments. For every mixture set, PR has the best separating power. This shows the merit of preserving concepts and selecting strong visual clusters during the clustering process.

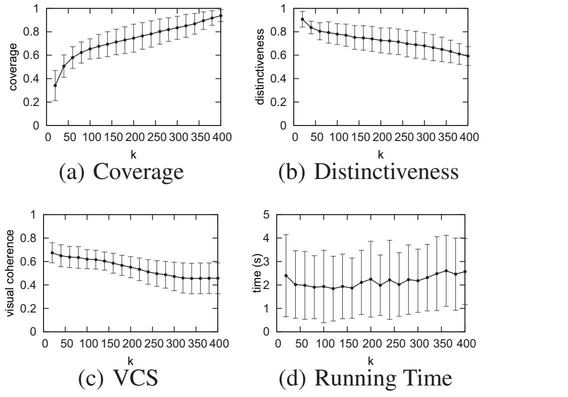
Comparison by evaluation metrics. Here, we aim to evaluate summarization methods based on the following evaluation metrics: visual coherence (VCS), coverage, distinctiveness and concept preservation. Figure 9(a) shows the scores of the summaries generated using the tested queries. The results indicate that AP and CV has superior VCS, coverage and distinctiveness compared to our method. This is unsurprising given that these methods are unconstrained by concept and cluster images purely on their visual similarities. Furthermore, they construct a partition on G, thus their perfect coverage and distinctiveness scores. However, this comes at a cost of low concept preservation scores, implying that association between a concept and a cluster is weak. On the other hand, the HY method has better concept preservation score, although in this case both VCS and concept preservation scores are inferior to PR. In summary, PRISM achieves the best balance of maintaining concept preservation and visual coherence of a summary. Using PR, the p-value in t-test against each method/metric is < 0.0001.

5.3 Analysis of PRISM

Effects of k. The parameter k controls the number of concept subgraphs of G in the decomposition. Figure 10(a) shows the effect of k on summary coverage with different result set sizes for all queries. Observe that the coverage of summaries increases with increasing k. At the same time, from Figure 10(b), the distinctiveness of summaries reduces with k. Figures 10(c) and (d), on the other hand, show that VCS reduces with increasing k values, while running time remain largely unaffected by k.

The results show that k controls the trade-off between summary distinctiveness and coverage. Unlike clustering methods that form a clustering that partitions the image set (*e.g.,* AP and CV), the need for concept preserving clusters imply that not all summaries constructed using our approach can achieve perfect uniqueness (distinctiveness) or representativeness (coverage). Often, to achieve maximum coverage, a certain amount of redundancies have to be allowed for, by creating overlapping concept-preserving clusters. Likewise, to achieve maximum distinctiveness, some images may have to be omitted because they could not be represented as non-overlapping concept-preserving clusters.

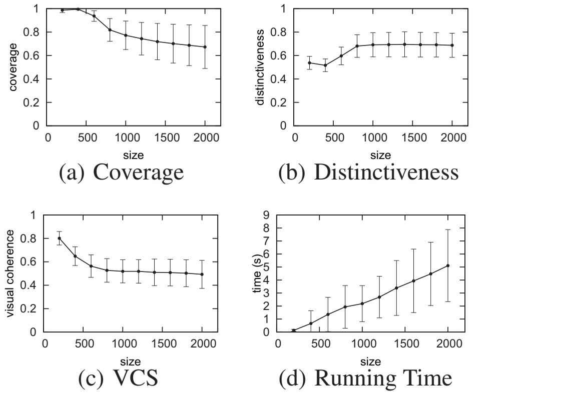
Effects of summary compression. Next, we study the importance of the summary compression phase using a user study. For



(a) Coverage (b) Distinctiveness

(c) VCS (d) Running Time

Figure 10: Effect of *k*.



(a) Coverage (b) Distinctiveness

(c) VCS (d) Running Time

Figure 11: Robustness of PRISM (at $k = 250$).

each summary, we performed 0% to 100% summary compression and evaluated the quality of each summary. We say that 100% compression is achieved when the summary cannot be compressed further. If we assume that the number of iterations needed to achieve 100% compression is n, then the m% percent compression is simply the summary after $mn/100$ compression iterations. Table 1 shows the *compression ratio* (computed as $|\mathcal{S}_0|/|\mathcal{S}_n|$) achieved for each tested query at 100% compression (numeric values associated with each query). Observe that our summary compression phase reduces the set of concept subgraphs for every query (up to a factor of 2.7). Next, for each query, assessors are presented a set of summaries with varying summary compression from 0% to 100% and requested to evaluated the visual appeal, relevance, comprehensiveness and organization quality of the summaries. Figure 9(b) reports the results. We observe that summary compression increases the perceived relevance of the summary. Summary is also seen as being better organized and more visually appealing. Similar to the effects of exemplar tag sparsity, summary compression reduces the complexity of the summary to create a more interpretable visual landscape of the query images. However, this comes at a cost of reduced perception of summary comprehensiveness. Nevertheless, the benefits gained on three other summarization qualities outweigh this loss of comprehensiveness.

Robustness of PRISM. We now investigate the robustness of PRISM to varying queries and result set sizes. We set $k = 250$ and study the distinctiveness, coverage and visual coherence of summaries for different queries and result sizes. Figures 11(a)-(c) show the results of the study. We observe that the error bars are small enough to justify that the summary quality is robust for varying result sizes.

Running time. Lastly, Figure 11(d) plots the running time of PRISM at varying result sizes. The error bars represent the standard deviation among different queries. The running time of PRISM scales relatively well with result size. Generally for top-1000 images, summarization can be completed in less than 3 seconds including construction of the visual similarity graph.

6. CONCLUSIONS

The quest for high quality social image search results visualization has become more pressing due to explosive growth of social image sharing platforms and search engines. In this paper, we have introduced three desirable features of a good social image search results summary, namely, concept-preservation, visual coherence, and coverage. We present a novel algorithm called PRISM which meets these desirable features. Specifically, PRISM utilizes both visual and concept features to construct a concept-preserving summary by refining, selecting, and compressing concept subgraphs. Based on this, an exemplar summary is easily created by selecting one or more exemplar image from each concept subgraph. Our empirical study demonstrated that PRISM produces superior quality summaries compared to state-of-the-art summarization techniques.

Acknowledgements: Boon-Siew Seah and Aixin Sun were supported by Singapore MOE AcRF Tier-1 Grant RG13/10.

7. REFERENCES

[1] M. B. Blaschko and C. H. Lampert. Correlational spectral clustering. In *IEEE CVPR*, 2008.

[2] D. Cai, et al. Hierarchical clustering of WWW image search results using visual, textual and link information. In *ACM MM*, 2004.

[3] H. Cheng, et al. Clustering large attributed information networks: an efficient incremental computing approach. *Data Mining and Knowledge Discovery*, 25(3), March 2012.

[4] T-S. Chua, et al. NUS-WIDE: a real-world web image database from National University of Singapore. In *ACM CIVR*, 2009.

[5] V. Chvatal. A Greedy Heuristic for the Set-Covering Problem. *Mathematics of Operations Research*, 4(3):233–235, 1979.

[6] B. J Frey and D. Dueck. Clustering by passing messages between data points. *Science*, 315(5814):972–976, 2007.

[7] Y. Jia, et al. Finding image exemplars using fast sparse affinity propagation. In *ACM MM*, 2008.

[8] Y. Jing and S. Baluja. VisualRank: applying PageRank to large-scale image search. *IEEE PAMI*, 30(11):1877–90, November 2008.

[9] Z. Li, et al. Grouping WWW Image Search Results by Novel Inhomogeneous Clustering Method. In *Proc. IEEE International Multimedia Modelling Conference*, 2005.

[10] P.-A. Moëllic, et al. Image clustering based on a shared nearest neighbors approach for tagged collections. In *ACM CIVR*, 2008.

[11] T. Opsahl and P. Panzarasa. Clustering in weighted networks. *Social Networks*, 31(2):155–163, 2009.

[12] R. Raguram and S. Lazebnik. Computing iconic summaries of general visual concepts. In *CVPR Workshops*, 2008.

[13] M. Rege, M. Dong, and J. Hua. Graph theoretical framework for simultaneously integrating visual and textual features for efficient web image clustering. In *ACM WWW*, 2008.

[14] B. S. Seah, et al. Summarizing Social Image Search Results. In *ACM WWW*, 2014.

[15] I. Simon, et al. Scene Summarization for Online Image Collections. In *IEEE ICCV*, 2007.

[16] A. Sun, S. S. Bhowmick, et al. Tag-based social image retrieval: An empirical evaluation, *JASIST*, 62(12), 2011.

[17] B. Q. Truong, et al. CASIS: a system for concept-aware social image search. In *ACM WWW*, 2012.

[18] R. H. van Leuken, et al. Visual diversification of image search results. In *ACM WWW*, 2009.

[19] S. Wang, et al. IGroup: presenting web image search results in semantic clusters. In *ACM CHI*, 2007.

[20] H. Xu, et al. Hybrid image summarization. In *ACM MM*, 2011.