Exploitation and Exploration Balanced Hierarchical Summary for Landmark Images

Jia Chen, Qin Jin, Shenghua Bao, Zhong Su, Shimin Chen, and Yong Yu

***Abstract*—While we have made significant progress over image understanding and search, how to meet the ultimate goal of satisfying both exploration and exploitation in one single system is still an open challenge. In the context of landmark images, it means that a system should not only be able to help users to quickly locate the photo they are interested in (exploitation), but also to discover different parts of the landmark which have never been seen before (exploration), which is a common request as evidenced by many recent multimedia studies. To the best of our knowledge, existing systems mainly focus on either exploration (e.g., photo browsing) or exploitation (e.g., representative photo identification), while users' need of exploration and exploitation is dynamically mixed. In this paper, we tackle the challenge by organizing landmark images into a hierarchical summary which gives user the flexibility of conducting both exploration and exploitation. In the hierarchical summary construction, we introduce two principles: the coherence principle and the diversity principle. Behind these two principles, the intrinsic concept is "detail-level," which measures how much detail that an image reflects for a certain landmark. A new objective function is derived from the definition of both exploration and exploitation experience on detail-level. The problem of finding an optimal hierarchical summary is formulated as searching over a space of trees for the one that achieves the best objective score. Extensive quantitative experimental results and comprehensive user studies show that the optimized hierarchical summary is able to satisfy both experiences simultaneously.**

***Index Terms*—Content-based retrieval.**

Manuscript received November 04, 2014; revised March 22, 2015 and June 11, 2015; accepted July 14, 2015. Date of publication July 23, 2015; date of current version September 15, 2015. This work was supported in part by the Fundamental Research Funds for the Central Universities and the Research Funds of Renmin University of China under Grant 14XNLQ01, in part by the Beijing Natural Science Foundation under Grant 4142029, in part by the NSFC under Grant 61303184, and in part by the Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Sheng-Wei Chen.

J. Chen and Y. Yu are with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: fchenjia@apex.sjtu.edu.cn; yyug@apex.sjtu.edu.cn).

Q. Jin is with the School of Information, Renmin University of China, Beijing 100872, China (e-mail: qjin@ruc.edu.cn).

S. Bao is with the IBM Watson Group, San Jose, CA 95120 USA (e-mail: baoshhua@us.ibm.com).

Z. Su is with the IBM China Research Laboratory, Beijing 100193, China (e-mail: suzhong@cn.ibm.com).

S. Chen is with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China (e-mail: chensm@ict.ac.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2015.2460111

I. Introduction

More and more image related applications, such as photo sharing web site Pinterest and image search engine Bing, provide both trending image function for exploration and keyword search function for exploitation. As defined in work [1], user's exploitation need is to specify a detailed and exact requirement while user's exploration need is to communicate to the system his/her broad categories of interest. Exploitation need in image related applications has been studied for years. Research works focusing on this type of need include near duplicate image retrieval [2], image feature hashing [3] and product image retrieval [4]. In recent years, exploration need in image related applications has attracted more and more research attention. Research works focusing on this type of need include faceted exploration [5], visual diversification [6] and trend image suggestion [7]. Balancing exploitation and exploration experience has also been studied, but it is usually limited in the relevance feedback of image retrieval [1].

In the domain of landmark images, a user may start browsing images without a specific objective (exploration behavior). In the process, he/she gets a general idea of the landmark and turns to extract relevant images of the landmark (exploitation behavior). Finally he/she may end up with a wanted photo or an experience of touring a landmark. Or he may start searching an image with a specific objective (exploitation behavior). In the process, he may discover some interesting details and turns to explore different details (exploration behavior). Such mixture behavior of exploitation and exploration has been noted in landmark image collection browsing [8]. However, most current research works on landmark images focus on satisfying only one kind of need. A typical solution to address the exploitation need is representative view [9]–[10], which generates images of canonical views but loses the spatial relation in such a result. Thus it is difficult for users to do smooth exploration. Another solution is StreetView [11] which locates photos on the map for users to browse photos along streets, but users usually don't know the location of the wanted photo beforehand. Thus it is difficult for users to do quick exploitation. To the best of our knowledge, there is no research work focusing on both types of experience on landmark images.

In this paper, we propose to satisfy these two types of user needs from a new point of view: organizing the landmark images in a hierarchical summary based on a tree structure. To be specific, we reduce exploitation experience to search operation along a path from root to leaf and reduce the exploration experience to tree traversal operation. Such hierarchical summary may lead to not only improved landmark image search result

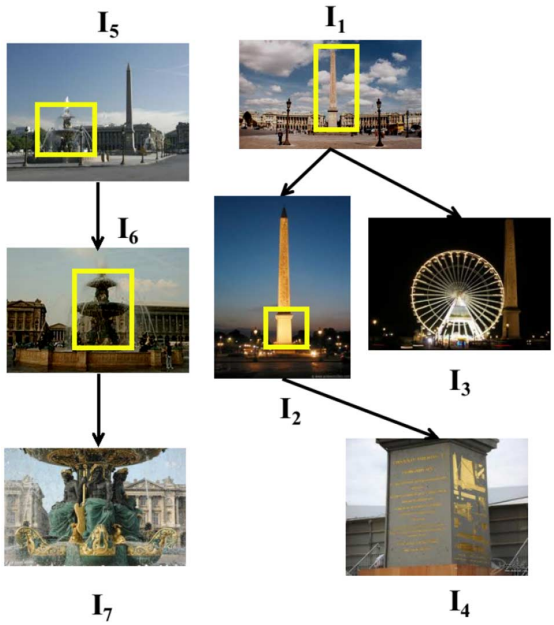


Fig. 1. Illustration example of hierarchical summary of landmark "Place de la Concorde": yellow rectangles in I_1, I_2, I_5, I_6 indicate the areas that have closer lookup in child nodes. I_1 and I_5 are the whole scene of the square. I_2 shows the Obelisk of Luxor. I_3 shows the ferris wheel near the square. I_4 shows the notes on the obelisk. I_6 and I_7 show the Fountain of River Commerce and Navigation in the square. I_8 shows the Church of the Madeleine behind the square.

but also better landmark representation. We want to point out here that it is not a visualization technique since we focus on organizing and summarizing the content rather than organizing the layout. We investigate the following two research questions step-by-step.

R1—What Does a Hierarchical Summary Look Like if it Meets Both Exploitation and Exploration Needs?: We design two principles for the hierarchical summary. The first one is the coherence principle: a child image matches and amplifies a part of the parent image. That is, the child image is more detailed than a parent image. The second one is the diversity principle: different parts of a parent image are matched and amplified by different children. That is, different children are about different details.

The coherence principle ensures that users can follow the change from whole scene to close detail along the path. The diversity principle ensures that users will explore different details if they follow different branches. Both principles are about relative detail relation between a parent image and a child image, but they describe such relation from different perspectives. Furthermore, we introduce "detail-level" concept to describe the absolute detail position of an image in the entire landmark image corpus. It measures how much detail that an image reflects for a certain landmark. Larger detail-level indicates that the image is more about close detail while smaller detail-level indicates that the image is more about whole scene. An example of hierarchical summary is given in Fig. 1.

In exploitation experience, users choose the right child from several children to continue search and prefer a shorter search path from the root to the wanted image node. In exploration experience, users prefer smoother change in traversal on images.

R2—Based on the Definition of Hierarchical Summary in R1, How to Construct Such a Summary? Constructing a hierarchical summary faces three challenges. The first challenge is that we need to estimate the detail level of each image, which is used in the description of exploitation and exploration experience. The second challenge is that the size of entire candidate hierarchical summaries (space of trees) is too large, which grows exponentially with the number of images. The third challenge is how to search the best solution in the space of trees.

Searching in space of trees for the optimal solution in acceptable time limit is far from trivial. To accelerate the search process, we build the graph base to prune the space of trees of candidate hierarchical summaries. We encode trees in the pruned space of trees by edge contraction and edge removal operations and design a transition function on the code to transfer between candidates. Finally we exploit simulated annealing framework to search the best solution in the pruned space of trees. Our algorithm runs less than 0.1s for the best solution on a space of trees composed of around 40 nodes.

Extensive experiments are conducted on the Oxford landmark dataset [12] and a Flickr landmark dataset of more than 500,000 images collected by our own. Altogether 304 worldwide landmarks are covered. Extensive evaluation are conducted on the proposed hierarchical summary. General statistics on the generated hierarchical summary shows that it contains abundant images with sufficient detail levels for high-quality exploration experience and moderate number of nodes for quick exploitation experience. Based on the manually labeled ground-truth, it is proved that the generated hierarchical summary has path orderliness property, branch diversity property and path smoothness property. User study shows that hierarchical summary is preferred by users over the traditional clustering based flat summary.

In summary, this paper makes the following contributions:

1) we propose two principles to organize landmark image corpus in a hierarchical summary to satisfy both exploitation need and exploration need;
2) we solve three challenges in constructing a hierarchical summary: detail-level estimation, pruning space of trees and searching over space of trees; and
3) we do extensive user studies to show the potential usefulness of our proposed hierarchical summary.

The rest of the paper is organized as follows. Section II presents related work. Section III introduces three properties of the hierarchical summary. Section IV formulates the problem. Section V designs the solution to tackle the three challenges of the problem. Section VI evaluates the proposed hierarchical summary. Section VII draws some conclusions.

II. Related Work

User's exploitation need has been recognized by the multimedia community for years. For content-based image retrieval, there have been abundant researches such as work [3], [13]. In work [13], they focus on building a better index and adding spatial verification as a post-process step to provide more accurate result in less time. Work [14] proposes a coupled Multi-Index (c-MI) framework to perform feature fusion at indexing level. Researches [3]–[16] focus on building different hash schemes

to embed high-dimensional features or complex distance functions into a low-dimensional hamming space where items can be efficiently searched. However, exploitation is only one kind of need in image search and we leverage exploitation techniques such as indexing to speed up summary construction in our work. In recent years, user's exploration need have attracted more and more researchers [7]–[6]. Work [6] diversifies the image search result by several clustering techniques based on image content. Work [5] provides keyword based faceted exploration for image search results. The objective of this system is to aggregate the knowledge and high-quality content that is available on and off the network, and to support the user in their quest for information by identifying the most relevant aspects of a query. Work [7] facilitates user's exploration experience by personalized image search. There are also researches focusing on balancing exploitation and exploration experience. Most of these researches are focused on the balancing issue in relevance feedback of image retrieval. Work [1] proposes an adaptive exploration/exploitation trade-off that transforms the original framework into a versatile retrieval framework with full searching capabilities. We focus on automatically balancing the exploitation and exploration experience by introducing the diversity constraint which is widely used in summarization tasks.

In the domain of landmark images, most related researches are works [8]–[9] focused on landmark photo exploitation while work [18] is focused on landmark photo exploration. Among exploitation focused works, their common solution style is to extract canonical photos. Among exploration focused works, their common solution style is photo collection navigation in 3D space. In the first category, researchers retrieve a few canonical photos and rank the results to meet exploitation need. Some researchers propose generating diversified image search result for landmark query, e.g. Lyndon Kennedy *et al.* [9], Ian Simon *et al.* [8], Stevan Rudinac *et al.* [17] and Junfeng Ye *et al.* [10]. Work [9] uses location and other metadata, as well as tags associated with images, and the images' visual features to generate a representative set of images for a landmark, which is actually a flat image summary without spatial connections among images in the summary. Work [8] examines the distribution of images in the collection to select a set of canonical views to form the scene summary, using clustering techniques on visual features, which is still a flat image summary. Work [17] proposes a novel edge weighting mechanism and a simple but effective scheme for selecting the most representative and diverse set of images based on the fusion of different kinds of information. Again it is a flat image summary without spatial connections. Work [10] presents a flat photo summary for each returned landmark, considering both visual representativeness and diversity. All these researches solve the diversity issue for exploitation experience to some extent but loses the spatial connection among the images in the result. This results in poor exploration experience. However, our exploitation and exploration balanced summary preserves not only the canonical photos but also the space connection among these photos.

In the second category, researchers focus on building a photo navigation system based on 3D reconstruction. In Noah Snavely *et al.*'s work [18]–[20], the big picture is represented as 3D point cloud which is usually sparse. To see details, user could choose photos from different cameras. It offers fantastic exploration experience but very poor exploitation experience. Since users don't know where is the wanted photo in 3D space beforehand, the exploitation efficiency in such model relies on the speed of manually locating/searching a photo in 3D location. Furthermore, their 3D model is represented by a sparse 3D point cloud rather than a 3D surface. Differently, users can quickly locate the wanted image in our hierarchical summary under the guidance along a path from the root. In summary, each category of existing work solves a paticular type of user experience very well but doesn't satisfy the other type of user experience well. However, these two types of experience are often mixed when users are browsing and searching landmark images. Our solution satisfies both types of user experience simultaneously. Work [21] uses scene map and geo-clustering to accelerate the retrieval process based on GPS data. A scene map preserves the spatial relation in a group of images with similar views and geo-clusters preserve spatial relation between different scene maps. However, their approach relies heavily on geo-tagged data. Our solution is purely based on vision data and can be possibly enhanced by geo-tagged data.

Another related field is landmark recognition. Work [22] proposes to leverage 3D visual phrases, a triangular facet on the surface of a reconstructed 3D landmark model, to improve recognition performance. Work [23] proposes to send compressed (low resolution) images to remote server instead of computing image features locally for landmark recognition and search. Work [24] proposes a new bag-of-visual phrase (BoP) approach for mobile landmark recognition based on discriminative learning of category-dependent visual phrases. These researches focus on recognizing the landmark from images taken at different view points. Landmark recognition can be used as a pre/post process stage for our work to filter out irrelevant or nearby landmark images.

III. Properties of Hierarchical Summary

Our proposed hierarchical summary follows a tree structure. It is generated by recursively applying the coherence and diversity principles as defined in Section I. From these two principles plus balancing exploitation and exploration objective, we directly induce three properties of the hierarchical summary. The first two properties are induced from the two principles respectively and the last property is induced from the balancing objective.

Path Orderliness Property: According to coherence principle, a child image in the hierarchical summary should register a part of the parent image. Thus, the detail-levels of nodes are sorted on a path from root to leaf. That is, images close to the root side are usually whole scene images (small detail-level value) while images close to the leaf side are likely to be close detail images (large detail-level value). In Fig. 1, the two root nodes (I_1 and I_5) are whole scenes of the square. The detail levels of images are sorted on all the 4 paths: $I_1 \rightarrow I_2 \rightarrow I_4$, $I_1 \rightarrow I_3$, $I_5 \rightarrow I_6 \rightarrow I_7$. This property helps to determine whether to go further along a path in exploitation experience. It also helps to provide order for exploration experience in the same path.

Branch Diversity Property: According to the diversity principle, children images branch into different spatial details of the

TABLE I
NOTATION TABLE

$\mathcal{T}_I$	the space of all possible hierarchical summary trees on image corpus I
$t=(V_t,E_t)$	V_t is the set of nodes in t and E_t is the set of edges in hierarchical summary tree t
$U_{exploit}(t)$	exploitation utility function of hierarchical summary tree t
x_n	detail level of node n in the hierarchical summary tree
d_n	depth of node n in the hierarchical summary tree
$U_{explore}(t)$	exploration utility function of hierarchical summary tree t
x_p, x_c	detail level of parent p and child c in edge e
x_b	detail level of bin b
α	parameter to combine $U_{exploit}(t)$ and $U_{explore}(t)$
$D(t)$	diversity of hierarchical summary tree t
d	predefined diversity value

parent in the hierarchical summary. The children of image I_6 branch into an image (I_7) about detail of the fountain in the square. The children of image I_2 branch into an image (I_3) about the ferry wheel near the obelisk and the other (I_4) about the detail of notes on the obelisk. It helps to determine to go along which path in exploitation experience. It also helps to provide a diversified exploration since the parent node works as an anchor node when path changes.

Path Smoothness Property: Along paths from root to leaves, images change smoothly from whole scene to close detail. Unlike path orderliness property, exploitation and exploration experience require smoothness at different intensities. In exploitation, users prefer sharper change so that they can reach the wanted image with fewer steps. In exploration, users prefer mild change so that they have smoother change along a path. We have to make a balance on this property for the two experiences.

IV. PROBLEM FORMULATION

The intrinsic concept behind the two principles of hierarchical summary is "detail-level", which goes beyond the relative detail relation between parent and child to the absolute value of image's detail position in the entire landmark image corpus. We use detail-level to measure how much detail that an image reflects for a certain landmark. Larger detail-level means that the image is more about close details. The computation of detail-level for an image will be described in detail in Section V-B. The input of the problem is an image corpus $\mathcal{I}$ of size n under the same landmark, where n is usually a very large number above 1,000. The image corpus under the same landmark can be collected from photo sharing website of images tagged with the landmark name, or from image search engine retrieval by the landmark name. The image corpus is usually large enough to cover different detail-levels of a landmark. Based on $\mathcal{I}$, we denote the space of all possible hierarchical summary trees as $\mathcal{T}_I$, whose size grows at least exponentially with the number of images.

For a hierarchical summary tree $t=(V_t,E_t)\in\mathcal{T}_I$, we characterize exploitation and exploration experience by two utility functions $U_{exploit}$ and $U_{explore}$ respectively. Here V_t is the set of nodes in t and E_t is the set of edges in t. We summarize in Table I the notations used in this section: detail-level x_n, depth of node d_n and so on.

Exploitation Experience: users prefer to quickly reach any node (the wanted image). Users want to reach images with shorter path expansion on average. Thus the general exploitation experience is defined as the average exploitation experience on each node. For each node, we use x_n/d_n, the ratio between an image's detail level x_n and its depth level d_n in the tree, to characterize the exploitation experience on a single node n in the tree. Larger x_n/d_n on a single node means that the detailed image appear at a shallow level in the tree so that users can quickly retrieve a detailed image, not to mention distant shot images which are at much shallower depth given the path orderliness property. Maximizing this measurement over all nodes bias towards larger detail-level to depth ratios. Since the detail-levels of images are fixed to the utility function, the actual bias in the later optimization step is towards trees with smaller depth. Please note that it doesn't bias towards fewer nodes since adding a node with large detail value at shallow depth actually increases $U_{exploit}$. Putting all the above together, we get the formula for exploitation utility

$$U_{exploit}(t)=\left(\sum_{n\in V_t} x_n/d_n\right)/|V_t|. \tag{1}$$

Exploration Experience: users prefer smooth change between parent and children at all detail-levels. Smoother change means smaller detail-level difference between parent and children. Thus the general exploration experience is defined as average exploration experience on each edge. For each edge, we use x_p/x_c to characterize the exploration experience. Here, x_p is the detail-level of the parent node and x_c is that of the child node. Smaller x_p/x_c on a single edge means that smoother change from the parent image to the child one. To cover changes at all detail-levels, we split the detail-level to 5 equal bins. Each edge is categorized to one bin based on the detail-level of the child node. In each bin, we calculate weighted average of edges in that bin: $S_b=\frac{\sum_{e\in E_b}(x_px_c)^{1/2}(x_p/x_c)}{\sum_{e\in E_b}(x_px_c)^{1/2}}$, where each edge is weighted by $(x_px_c)^{1/2}$ since users are more sensitive to changes at closer detail-level than changes at distant shots. Putting all the above together, we get the formula for exploration utility

$$\begin{aligned} U_{explore}(t)&=\sum_{b\in 1,\cdots,5} x_bS_b\\ where\quad x_b&=\max_{e\in E_b}(x_c)\\ S_b&=\frac{\sum_{e\in E_b}(x_px_c)^{1/2}(x_p/x_c)}{\sum_{e\in E_b}(x_px_c)^{1/2}}. \end{aligned} \tag{2}$$

We introduce parameter α to combine the above two utility functions

$$U(\alpha,t)=\alpha U_{exploit}(t)+(1-\alpha)U_{explore}(t). \tag{3}$$

To achieve balance between exploitation and exploration with flexible α, we further introduce a diversity constraint (4) which requires the diversity of $D(t)$ of t in the solution to be

around a certain value d[1] within the range of the slack variable ξ. $D(t)$ is defined as the minimum of pairwise Euclidean distances on gist [25] feature between images in the summary

$$d - \xi \leq D(t) \leq d + \xi. \tag{4}$$

Diversity is an important criterion in both image summarization [26] and text summarization tasks [27] while gist is widely used in image summarization tasks [28]. In our task, if α is set to 0 (i.e. we only focus on exploration experience), the smooth changes on edges are small. This leads to many similar images in the summary and the diversity value will be small. On the other hand, if α is set to 1 (i.e. we only focus on exploitation experience), the changes on edges are dramatic. This leads to many images sharing little or none content and the diversity value will be large. A summary balancing exploitation and exploration experience should have a suitable diversity value.

Given the objective function, the tree space and the constraint, we get the formulation of our main problem.

1) Main Problem: On hierarchical summary tree space $\mathcal{T}_I$, finding an exploitation and exploration balanced hierarchical summary is to find the solution of the following optimization problem:

$$\begin{aligned}(\hat{t}, \hat{\alpha}) &= \arg \max_{(t,\alpha)\in\mathcal{T}_I\times[0,1]} U(\alpha, t) - C\xi \\ &= \arg \max_{(t,\alpha)\in\mathcal{T}_I\times[0,1]} \alpha U_{exploit}(t) \\ &\quad + (1-\alpha)U_{explore}(t) - C\xi \\ s.t. \quad & d - \xi \leq D(t) \leq d + \xi \\ & \xi \geq 0 \end{aligned} \tag{5}$$

where C is an extremely large positive constant to restrict the slack variable ξ since we don't want the constraint to be violated intensively.

In the solution $(\hat{t}, \hat{\alpha})$, we get the best hierarchical summary $\hat{t}$ and the best weighting of exploitation utility and exploration utility combination simultaneously. By considering the constraint as a penalty item $|D(t) - d| < \xi$ as in book [29], we can rewrite the main problem in the penalty form:

2) Main-Penalty Problem:

$$(\hat{t}, \hat{\alpha}) = \arg \max_{(t,\alpha)\in\mathcal{T}_I\times[0,1]} U(\alpha, t) - \lambda|D(t) - d| \tag{6}$$

where $\lambda > 0$ and it is an extremely large positive number to penalize contradiction of the constraint.

V. Solution

The main problem is non-convex and difficult to solve. To approximate the solution of the problem, we decompose it on solution space and adopt a divide and conquer approach. The solution space is the Cartesian product of the parameter α and the tree t. In the decomposition stage, we fix α, the balance between exploitation and exploration, and find the best t under such balance without considering the diversity constraint. This is denoted as α-problem. Once we get a list of best $\hat{t}_\alpha$s for different parameter αs, we select the t that best matches the diversity constraint in the conquer stage. Since for each α, $\hat{t}_\alpha$ is fixed in this stage, selecting the t that best matches the constraint equals to selecting a suitable parameter α. This is denoted as $\hat{t}_\alpha$ problem. In a summary, we divide the problem into a set of α problems[2] and merge solutions $\hat{t}_\alpha$ by the diversity constraint in the $\hat{t}_\alpha$ problem. Similar to other non-convex problems, our approach only finds a near optimal solution. We formulate α-problem and $\hat{t}_\alpha$ problem as follows.

1) α-problem: Given α fixed, we search over the space of trees $\mathcal{T}_I$ for the tree candidate which maximizes the combined utility function $U(t; \alpha)$

$$\hat{t}_\alpha = \arg \max_{t\in\mathcal{T}_I} U(t; \alpha). \tag{7}$$

We write the combined utility function $U(\alpha, t)$ in the form $U(t; \alpha)$ to emphasize that α is given.

2) $\hat{t}_\alpha$-problem: Find the best $\hat{t}$ *among the solution set* $\{\hat{t}_\alpha\}$ of α-problems

$$\hat{t} = \arg \max_{t\in\{\hat{t}_\alpha\}} U(\alpha, t) - \lambda|D(t) - d|. \tag{8}$$

$\hat{t}_\alpha$-problem is easy to solve while α-problem has three challenges. First, the detail-level of each image x_i in $U(t; \alpha)$ is unknown. Second, the size of the possible space of trees $\mathcal{T}_I$ grows at least exponentially with the number of images. Third, most well-known searching algorithms don't work on the space of trees. Before giving solutions to the three challenges sequentially, we build a match graph for the image corpus as the start point of our solutions. The match graph captures the relative detail relationship between images. The flowchart of the solution is summarized in Fig. 2.

A. Match Graph

The relative detail relation in the two principles is based on image matching operation. Thus we build a match graph to capture relative detail relation between images in the landmark image set. We calculate spatial transform to measure the matching quality of image pairs following the standard procedure [12]: we extract SIFT features [30] on image pairs and filter the matching by RANSAC algorithm [31] to estimate a 4 degree of freedom(dof) homography transform $\mathbf{H}$: $[s_x, 0, t_x; 0, s_y, t_y; 0, 0, 1]$.[3] s_x and s_y are horizontal and vertical scale changes respectively while t_x and t_y are horizontal and vertical translations respectively. To reduce the computation cost, we filter out image pairs whose cosine similarity on Bag of Words (BoW)[4] is below the threshold 0.1 before matching images. We generate connected components by connecting image pairs that have more than 8 matched points after tuning. All connected components together form the match graph. On the match graph, the exploration experience is ensured since it

[1]In implementation, d is set to 0.1

[2]The sampling step of α is set to 0.01 in implementation.

[3]We don't consider rotation and shear transforms since they rarely occur in landmark images.

[4]The word number is set to 20 000 after considering the balance between effect and computational cost.

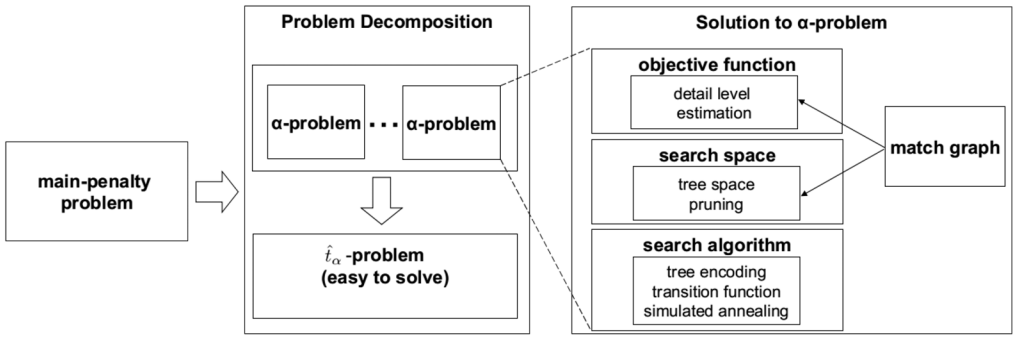


Fig. 2. Flowchart of our solution. The main-penalty problem is decomposed into two sub-problems: solve a set of α-problems, and then solve $\hat{t}_\alpha$-problem. $\hat{t}_\alpha$-problem is relatively easy and α-problem has three challenges. We estimate the detail-level, which is used in the objective function. We pruned the space of trees, which grows exponentially. We design a tree encoding and transition function to utilize simulated annealing for searching over space of trees. We build a match graph for the image corpus, which is the start point of our solutions.

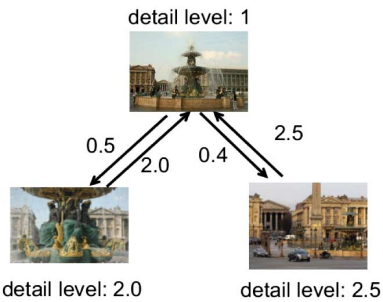


Fig. 3. Flow graph. The values on the edges are pairwise ratios.

encodes image matching information. In the following subsections, it is not only used to estimate detail-level from the match graph but also used to cut down the tree space. In this graph, images can be transformed to at least one other image within the same connected component. Thus a connected component in the match graph naturally forms a view division. Small connected components are usually composed of irrelevant photos and we discard components whose size are smaller than 5. On each connected component, we build a hierarchical summary and the final hierarchical summary of the landmark is built by aggregating these hierarchical summaries by a virtual root.

B. Detail Level Estimation

Directly from the matching information, we know the scale ratio between image pairs. The scale ratio is calculated by $(s_x s_y)^{1/2}$, which is the geometric mean of horizontal scaling s_x and vertical scaling s_y from $\mathbf{H}_{4dof}$. However, we don't get the general knowledge about the number of photos at different detail-levels. We design an algorithm to estimate the detail-level of all the images from the matching information.

Flow Graph and Flow Matrix: We construct the flow graph by linking each matched image pair with two directed edges and assign each edge with the scale ratio from the head node image to the tail node image. The weight on the edge represents the amount of flow from the head node to the tail node in each round. On each node, there is a reservoir to store the accepted flow in each round. The amount of accumulated flow in the reservoir is the detail-level of the corresponding image. An illustration of the flow graph is given in Fig. 3. The general updating rule of the values on nodes can be written as $\mathbf{x} = \mathbf{F}\mathbf{x}$, where $\mathbf{x}$ is the detail-level vector of all nodes and $\mathbf{F}$ is the flow matrix. An image at larger detail-level receives more flow from its neighbors since the edges ending at it have larger weights. At last, images of close detail accumulate far more flow than the images of whole scene.

We construct the flow matrix from the flow graph's adjacency matrix $\mathbf{A} = [(\mathbf{A}_1)^\top, \cdots, (\mathbf{A}_n)^\top]^\top$, where $\mathbf{A}_i$ is the ith row of $\mathbf{A}$. Recall that element $\mathbf{A}_{ij}$ is the scale ratio from photo i to photo j. Thus we have $\mathbf{A}_{ij} = 1/\mathbf{A}_{ji}$ and $\mathbf{A}_{ii} = 1$. In each round, the flow from node j to node i is calculated by $x_j * \mathbf{A}_{ij}$, the product of the scale ratio and the accumulated flow in the reservoir of node j. We update the detail-level vector by the average of flows contributed from its neighbors. That is, flow matrix $\mathbf{F}$ is obtained from normalizing the adjacency matrix $\mathbf{A}$ by the number of non-zero elements in each row

$$\mathbf{F} = [\mathbf{A}_1^\top / ||\mathbf{A}_1||_0, \cdots, \mathbf{A}_n^\top / ||\mathbf{A}_n||_0]^\top. \tag{9}$$

After sufficient rounds of propagation, vector $\mathbf{x}$ converges to the eigenvector associated with the largest eigenvalue of the flow matrix $\mathbf{F}$. A standard proof in numerical matrix computation [32] will confirm this convergence.

C. Pruning Space of Trees

To prune space of trees, we start from match graph G_m, followed by 5 key steps: 1) we extract representative images; 2) we transform G_m into representative graph G_{rep} by preserving only representative images; 3) we denote each connected component in the representative graph as a graph base; 4) we get several potential tree bases from the graph base by different edge removal operations; and 5) on each tree base, we apply edge contraction operation to get hierarchical summary candidates, which is the pruned space of trees. Fig. 4 illustrates this process and details are described below.

1. Representative image: we get representative images by applying affinity propagation [33] to find out exemplars as representative images. The similarity between images are set to the negative of their Euclidean distance on gist feature. In affinity propagation, we need to set the preference value[5] rather than the cluster number.

[5]The preference value is set to -1.0

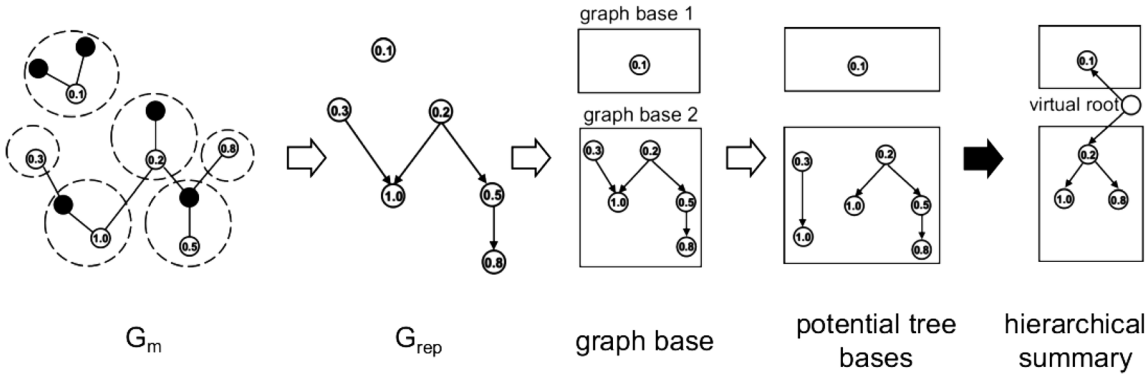


Fig. 4. Illustration of pruning space of trees: match graph → representative graph → graph base → potential tree bases. Then hierarchical summary is extracted on the pruned space of trees. The white nodes are representative images and the black ones are not. The dashed circles indicate which black nodes are represented by the white node. The numbers in the nodes are detail-levels.

2. $G_m \rightarrow G_{rep}$: from match graph $G_m = (V_m, E_m)$, we apply vertex contraction operation to nodes that are not representative images. Here V_m is the set of nodes in G_m and E_m is the set of edges in G_m. Then, we remove multiple edges from this graph. The result graph is an undirected graph and we turn it into a directed graph by adding directions from nodes with smaller detail levels to nodes with larger detail levels. The representative graph $G_{rep} = \{V_{rep}, E_{rep}\}$ is a directed acyclic graph as illustrated in Fig. 4.
3. $G_{rep} \rightarrow$ graph bases: we define each connected component in G_{rep} as a graph base.
4. Graph base → tree bases: now the only difference between the graph base and a tree is that in the graph base a node may have more than one parents. We denote such nodes as key nodes K. We turn a graph base into a tree base by preserving only one parent on each key node. Based on the combination of parent preservation choices, we get several potential tree bases from one graph base. We denote the set of potential tree bases from graph base g as $\mathcal{T}_g$.
5. Tree base → candidates: for each tree base in this set, there is no improvement space for better exploration experience since we have no node to add between parent and children in the tree base. But there is significant improvement space on exploitation experience. To achieve this goal, we introduce edge contraction operation $\mathcal{C}$, which helps us to merge images that have similar detail levels. When edge contraction operation happens, we always remove the parent node. The edge contraction operation can be applied multiple times until there is only one node in the resulting tree. Thus the pruned space of trees induced by the graph base is $S = \mathcal{T}_g \cup \mathcal{C} \circ \mathcal{T}_g \cup \mathcal{C}^2 \circ \mathcal{T}_g \cup \cdots$, where $\mathcal{C}^n \circ \mathcal{T}_g$ means the pruned space after n times of edge contraction operations. A hierarchical summary is a tree aggregating these directed trees by a virtual root in the pruned space of trees.

D. Searching Over Pruned Space of Trees

To search over the pruned space of trees, we encode trees by edge contraction and removal operations and design a transition function between trees based on their codes. Given the code and the transiton function, we utilize simulated annealing framework, which is often used when search space is discrete and more efficient than exhaustive search, to search the best solution in the space of trees. For detailed introduction of simulated annealing, we refer readers to work [34]. Here we focus on customizing the encoding scheme and transition function in our task.

Tree encoding: We encode the trees in the space of trees by the edges in the graph base. We set $11\cdots1$ for the graph base. Each bit corresponds to an edge in the graph base. For each tree in this space, we encode removed edges from the graph base to the tree base by 2 and contracted edges in the tree base by 0. Thus we get an encoding scheme for each tree in the space of trees. Different edge contraction combinations on different tree bases lead to different codes.

Transition function: Randomly flipping a bit in the code doesn't necessearily lead to a valid code of tree in the pruned space of trees. Thus we need to design a transition scheme on the code. The transition scheme is composed of two stages. In the first stage, we randomly change the tree base. To be specific, we have 50% chance to randomly change the parent of one key node and 50% chance to stay on the current tree base of the code. In the second stage, we randomly change one edge contraction state on the tree base of the code.

We utilize simulated annealing framework to search the best value of the objective function in the space of trees. In this framework, there are three things to customize: energy function, neighbors of a state and annealing schedule. For the energy function, we take the negative of the objective function $E(t) = -U(t)$. For the neighbors of a state, we use the transition function to generate valid neighbor states. For the annealing schedule, we use exponential annealing schedule. The entire algorithm is summarized in Algorithm 1.

VI. Experiment

In this section, we evaluate the generated hierarchical summary from the following three perspectives:

1) general statistics on the hierarchy structure;
2) quantitative evaluation on the three properties: path orderliness, branch diversity and path smoothness; and
3) user study on exploitation and exploration balancing

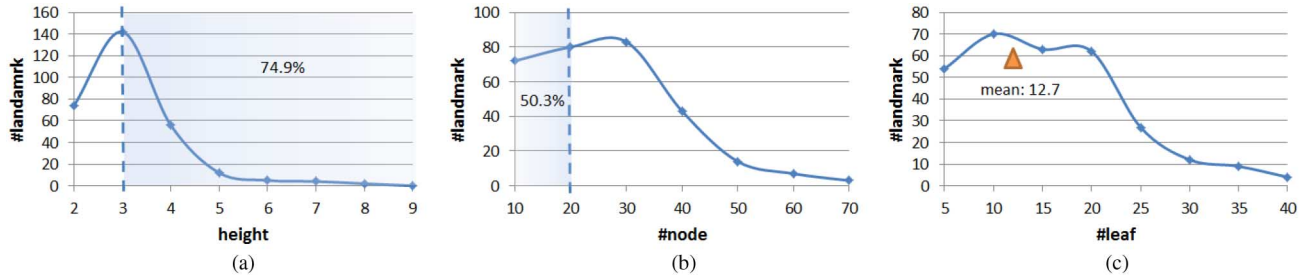


Fig. 5. Statistics of critical parameters in hierarchical summary. Medium height, no more than 20 nodes, and a relatively large number of leaves is preferred for a balanced hierarchical summary. (a) Height of the summary. (b) Nodes in the summary. (c) Leaves in the summary.

A. Experiment Dataset

We evaluate the proposed hierarchical summary on two datasets: *Oxford* dataset, which is publicly available [35] and *Flickr* dataset collected by our own, which is much larger and contains more noises. The *Oxford* dataset consists of 5,062 images collected from Flickr of 19 Oxford landmarks. On average there are 200 images for each landmark. Landmarks "new" and "oxford" are removed since they are not specific landmarks. We also remove three landmarks which have fewer than 100 images. The *Flickr* dataset is crawled from Flickr by landmarks in the list from paper [36]. From the list, we select 295 landmarks into the final *Flickr* dataset, each of which have more than 1,000 images.

B. General Statistics on Hierarchy Structure

The global quality of a hierarchy is reflected by the height, the number of nodes and the number of leaves. The height of a tree represents the detail-level number in the summary: larger height means more detail-levels in the summary. The number of nodes represents the size of the summary: larger number of nodes means that it is harder to see the entire summary at a glance. The number of leaves represents the amount of details in the summary: larger number of leaves means more details in the summary.

Algorithm 1: Simulated Annealing on space of trees

input: codes $\{c\}$ of trees in space of trees, the maximum number of rounds k_{max} in simulated annealing, the number of states S on each temperature
output: best code c_{best}
Randomly select a tree base as the initial state c;
$k \leftarrow 0$
$T \leftarrow T_0$
while $k < k_{max}$ **do**
  for $s \leftarrow 1$ **to** S **do**
    $c_{new} \leftarrow \texttt{trans}\ (c)$
    if $e^{-(E(c_{new})-E(c))/T} > rand()$ **then**
      $c \leftarrow c_{new}$
  $T \leftarrow \alpha T$
  $k \leftarrow k + 1$
trans
  input: code c
  output: valid neighbor code c_{new}
  if $rand() > 0.5$ **then**
    generate the tree base code by randomly change the parent of one key node;
    project the original code on the new tree base by AND operation: $c_{new} \leftarrow c \& c_{treebase}$;
  else
    $c_{new} \leftarrow c$
  randomly select an edge on the tree base
  flip the corresponding bit on c_{new};

Fig. 6. Leaves in summaries: abundant detail photos indicate high-quality exploration experience.

We plot histogram of these three quantities on 295 landmarks' summaries. In Fig. 5(a), we see that for 74.9% landmarks, hierarchical summary height is larger than 3, which indicates that most landmarks have photos of different detail-levels for high-quality exploration experience. In Fig. 5(b), we see that more than 50.3% summaries have less than 20 nodes while number 20 is the number of images shown on the first result page of a typical image search engine. That is, more than half hierarchical summaries can be seen at a glance. This indicates that our proposed hierarchical summary provides comparable exploitation experience on the number of images. In Fig. 5(c), each summary has 12.7 leaves on average, which indicates that abundant details are preserved. Examples of leaves in summaries are shown in Fig. 6. The number of leaves on average benefits both exploitation and exploration experience. For exploitation, close detail photos can be found and for exploration experience, users can explore different details of the landmark.

C. Evaluation on Properties of Hierarchical Summary

As introduced in Section III, the hierarchical summary have three properties: path orderliness, branch diversity and path smoothness. The path orderliness measures the orderliness of nodes' detail-level in a path. The branch diversity measures the diversity of children images from the same parent image. The

TABLE II
SORTEDNESS OF ESTIMATED DETAIL-LEVEL ON THE OXFORD DATASET AND FLICKR DATASET

Oxford dataset			Flickr dataset		
landmark	RUN	INV	landmark	RUN	INV
All Souls	2/50	6/1225	Pasha Bulker	4/50	131/1225
Bodleian	2/26	3/325	Schloss Pillnitz	3/25	17/303
Chrish Church	0/50	0/1225	Karlsplatz	1/48	0/1128
Magdalen	2/50	2/1225	Taormina Church	3/27	35/351
Pitt Rivers	1/7	0/21	Piazza delle Erbe	1/50	0/1225
Radcliffe Camera	3/50	5/1225	De Waag	1/50	0/1225
Trinity	1/15	0/105	National Monument	2/50	44/1225

path smoothness measures the smoothness of image change along the path from root to leaf.

Evaluation on Path Orderliness: The path orderliness property comes from the detail-level estimation accuracy. We uniformly sample images at different estimated detail-levels. At most 50 images are sampled for each landmark. We manually align these images along detail-level axis to get the sorted groundtruth. We use sortedness to evaluate the difference between the sorting of groundtruth and the sorting induced by the estimated detail-levels. To be specific, we use two metrics, RUN and INV, from paper [37] to measure such difference. RUN is the number of ascending substrings and INV is the number of inversion pairs. In both metrics, smaller values indicate better sortedness. Their definitions are shown in (10), where $X =< x_1, \cdots, x_n >$ and x_i is the image position in the groundtruth and i is the image position induced by the estimated detail-level

$$\begin{aligned} RUN(X) &= |\{i | 1 \leq i < n \text{ and } x_{i+1} < x_i\}| + 1 \\ INV(X) &= |\{(i,j) | 1 \leq i < j \leq n \text{ and } x_i > x_j\}|. \end{aligned} \quad (10)$$

For a list of size N, the maximum value of RUN is N and the minimum value is 1. The maximum value of INV is $N(N-1)/2$ and the minimum value is 0. The maximum value of RUN and INV is reached when the sequence is in the reverse order of the ground truth. Table II shows the sortedness on 7 landmarks from Oxford and Flickr respectively. In the RUN and INV column, the number before symbol "/" is the sortedness induced by estimated distance and the number after the symbol is the sortedness of the worst case(reverse order). On both datasets, we have near perfect sortedness on detail-levels under both RUN and INV metrics. We also give an illustration example of sortedness in Fig. 7. Images are arranged from close detail to distant shot uniformly. Such sortedness accelerates the search from root to a node in exploitation experience. It also provides order in exploration experience.

Evaluation on Branch Diversity: The branch diversity property comes from the parent-children group in the hierarchical summary. Examples of parent-children groups are shown in Fig. 8(c). We invite 5 volunteers to label the groundtruth whether children images match and amplify different parts of its parent image. If two of them match the same part of the parent image, this parent-children group will be labeled as false, otherwise as true. We sample 50 landmarks and there are altogether 179 non-trivial parent-children groups in these landmark summaries. A non-trivial parent-children group means that there are more than one child in the group.

We calculate a branch diversity score on each parent-children group by counting the percentage of true labels from the 5 volunteers. For example, if a parent-children group is labeled as true by 3 volunteers and as false by the other 2, the score of this group is 0.6. We then calculate the branch diversity score of landmark summary by averaging the branch diversity score of the parent-children groups in the same landmark. The result represents the average branch diversity quality of the parent-children pairs in the landmark. It ranges from 0 to 1 and 1 means that all the parent-children groups in this landmark are considered as diversified branching unanimously by all the 5 volunteers. We plot the distribution of branch diversity score on landmarks in Fig. 8(b). The average branch diversity score is 0.83 and 32 out of the 50 landmarks get score 1, which means that more than 60% landmark summaries receive unanimous affirmative votes from all the 5 volunteers for branch diversity. It can imply that users could quickly determine to search along which path in exploitation experience. It also maximizes flexibility in exploration experience.

Evaluation on Path Smoothness: The path smoothness property comes from the detail-level change along the paths. Different from path orderliness which is focused on the order of detail-level, this property is focused on the smoothness of changes. We invite 5 volunteers to label the groundtruth whether the detail changes are smooth in each path by 0 or 1. If one detail change is too sharp or too mild in the path, this path will be labeled as false. "Too sharp" means that the user can not find the spatial relation in the change, which indicates poor exploration experience. "Too mild" means that the user think these two images are near duplicate, which indicates poor exploitation experience. We sample 50 landmarks and there are altogether 224 non-trivial paths in these landmark summaries. A non-trivial path is a path whose length is more than 3.

We calculate a path smoothness score from each path by counting the percentage of true labels from the 5 volunteers. We then calculate the path smoothness score for each landmark by averaging the path smoothness score of the paths in the same landmark. The result path smoothness score represents the average smoothness of the paths in summary. It ranges from 0 to 1 and 1 means that all the paths are considered to have suitable smoothness by all the 5 volunteers. We plot the distribution of path smoothness score on landmarks in Fig. 8(a). The average path smoothness score is 0.66. 23 out of 50 landmarks get score 1, which means that near half landmark summaries receive unaimous affirmative votes from all the 5 volunteers for path smoothness. It can imply that the generated hierarchical summary reaches the balance between exploitation and exploration experience on path smoothness.

D. User Study on Exploitation and Exploration Balancing

User Background and Evaluation Metric: The user study is done on 10 landmarks investigating user experience of 18 volunteers. The name of the 10 landmarks are shown in Table III. Among the 18 volunteers, 12 of them come from 2 universities

Fig. 7. (left to right) Images change from close shots to distant shots. The estimated detail-levels are shown in the text below images and yellow rectangles on images are used to help readers follow the change across the images. The ordered detail-level change accelerates the search from root to node in exploitation experience.

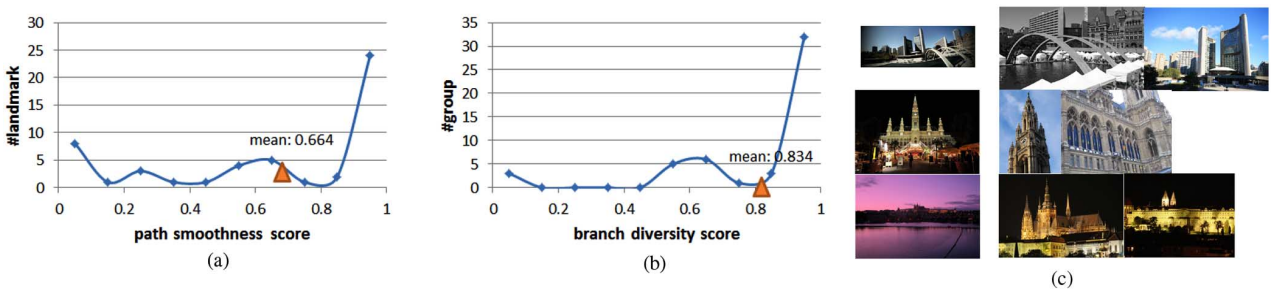


Fig. 8. Evaluation of path smoothness and branch diversity. The balance between exploitation and exploration experience is achieved on path smoothness; better exploitation experience and high-quality exploration experience is achieved on branch diversity. (a) Evaluation of path smoothness. (b) Evaluation of branch diversity. (c) Branch diversity examples: different parts in the parent image (leftmost image) are registered as different children images on the right.

TABLE III
10 LANDMARKS IN USER STUDY

L1	pasha bulker	L6	capitoline hill
L2	christ the redeemer	L7	morelia cathedral
L3	zheng yang men	L8	giralda
L4	hohenschwangau castle	L9	manhattan bridge
L5	red fort	L10	place de la concorde

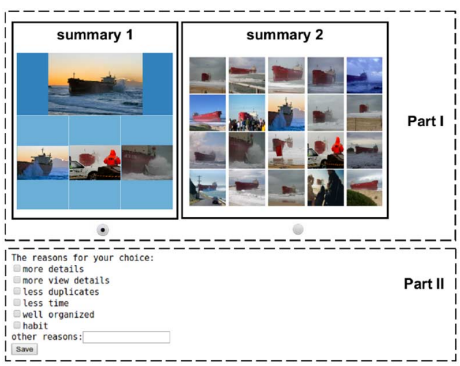


Fig. 9. Structure of the user study system interface. Users select the preferred summary in Part I and then select aspects in Part II.

and 6 of them come from 3 companies. Their age ranges from 20 to 60. 13 volunteers are male and 5 volunteers are female. Fig. 9 shows the structure of the user study system interface. The two compared summaries are shown side-by-side on the top of the screen. At the bottom of the screen shows possible reason candidates.

We require users to choose one of the summaries based on the following criterion:

The one showing more details or view points with fewer images is the better one: In this criterion, there are two contradictory parts: more details/view points and fewer images. Thus, it depends on the user to balance these two parts. However, it is difficult for human beings to follow this criterion consistently. In a comparison study, this criterion degenerates to a simpler one which is easier to execute.

1. *If the summary with more images don't contain more details that a user considers as interesting, the user will choose the one with less images and vice versa.*
2. *Users are not required to make a choice if it is too hard for her/him to judge which one is better.*

We use "interesting" in the criteria 1 to capture user satisfication/user need while user satisfication/user need could be subjective to some extent. We also define multiple reasons to let users select in the next step, which minimizes subjectiveness and is more objective. After making the choice of which one is better, users will choose one or more candidate reasons for their choice. This part is optional since sometimes users attribute their choices to feeling rather than to a particular reason. There are three groups of candidate reasons: 1. "more view points/details" aspect corresponds to "more view points" and "more details" reasons. 2. "less time/fewer duplicates" aspect corresponds to the original "less time" and "fewer duplicates" reasons. 3. "prefer the structure" corresponds to the original "well organized" and "habit" reasons. The first aspect is closely related with exploration experience. It also correlates to exploitation expericence since a summary with critical view point images missing does not provide good exploitation experience either. The second aspect is closely related with exploitation experience. It also correlates to exploration experience since a summary with duplicate images does not provide good exploration experience either. The third aspect is closely related with the structure organization of the summary. It is meaningful when comparing summaries of different structures. It covers both exploitation and exploration experience.

Fig. 10. Five baselines. (a) and (b) are the comparison to the first two baselines studies' real user feedback on exploitation and exploration experience; (c)–(e) are the comparison to the last three baseline studies on the effectiveness of an organized summary along detail-level. (a) B1. (b) B2. (c) B3. (d) B4. (e) B5.

Comparison to Five Baselines: We compare user experience on hierarchical summary with balanced objective to the following four baselines: 1) B1: hierarchical summary with exploitation objective only (α is set to 1 in main problem); 2) B2: hierarchical summary with exploration objective only (α is set to 0 in main problem); 3) B3: flat summary (affinity propagation clustering [33] on the match graph); 4) B4: hierarchy organization of flat summary by hierarchical clustering; and 5) B5: kmeans summary (a common baseline in image summarization [9], [17]). The first two baselines are focused on one type of user experience only as shown in Fig. 10(a) and 10(b). Comparison to these two baselines (comparison 1 and comparison 2 namely) not only shows the effectiveness of our solution $(\hat{t}, \hat{\alpha})$ to the main problem but also helps to study real user feedback on exploitation and exploration experience. The third baseline extracts representative images by affinity propagation clustering as shown in Fig. 10(c). Affinity propagation clustering is shown to work well on selecting representative images [33]. Comparison to the third baseline (comparison 3) shows the effectiveness of organized summary for image collection. The fourth baseline constructs a hierarchy structure on the third baseline by hierarchical clustering. An example is shown in Fig. 10(d). To be specific, we use complete linkage agglomerative clustering [38] and do not use detail-level information. Comparison to the fourth baseline (comparison 4) shows the superiority of the two principles and the importance of detail-level concept for hierarchical summary. The fifth baseline extracts representative images by kmeans, which is a common baseline used in summarization task [9], [17]. The cluster number is set to 10. We don't compare ours to work [9] and [17] directly since they require additional metadata while our approach is designed for more general landmark datasets where metadata is not available.

In comparison 1, 2, 3, 4, 5, more users select balanced hierarchical summary as the favorite one on 7/10 (7 out of 10), 7/10, 8/10, 9/10, 8/10 landmarks respectively. The case study of hierarchical summary as the favorite one is shown in Fig. 11. We also study failed cases in comparison 1 and 3. A failed case on L10 in comparison 1 is illustrated in Fig. 12(a). In the balanced hierarchical summary of L10, it is difficult for the users to find the relation between the two subtrees in it despite that the hierarchical summary contains an addition branch of landmark photos from a different point view compared to the exploitation objective summary. Thus, based on the evaluation criteria in Section VI-D1, they choose the exploitation objective summary rather than the hierarchical summary. A failed case on L3 in comparison 3 is illustrated in Fig. 12(b). Actually none of the summaries are ideal. On one hand, there are not enough images in the hierarchical summary for users to get an idea of the appearance of the landmark. On the other hand, there are some irrelevant images and similar ones in the flat summary. We conjecture that the users choose the flat summary might because they are more familiar with the flat structure.

Detailed analysis of five comparisons on the three aspects are listed in Table IV. In comparison 1, our method wins over B1 by 28 votes on "more view points/details" aspect. On the contrary, B1 wins over our method by merely 8 votes on "less time/fewer duplicates" aspect. This indicates that our method provides much better exploration experience and comparable exploitation experience compared to B1. In comparison 2, our method wins over B2 by 23 votes on "less time/fewer duplicates" aspect and ties with B2 (2 vote weak advantage) on "more view points/details" aspect. This indicates that our method provide much better exploitation experience and comparable exploration experience compared to B2. In comparison 3, our method wins over B3 by 27 votes on "prefer the structure" aspect. This indicates that the hierarchy structure does provide better over all experience compared to B3. In comparison 4, our method wins over B4 by 60 votes on "prefer the structure" aspect. On the contrary, B4 wins over our method by merely 11 votes on "more view points/details" aspect. This indicates that organizing hierarchy structure based on detail level is more welcomed by users compared to B4, which organizes hierarchy structure by hierarchical clustering. In comparison 5, our method wins over B5 by 38 votes on "prefer the structure" aspect. This indicates that the hierarchy structure does provide better over all experience compared to B5.

Discussion: If the landmark has many different details (e.g. sculptures on the wall) or it looks very different from different view points (e.g. the front and back of a cathedral), the hierarchical summary is preferred by most users. In such cases, the hierarchy structure of the summary has several branches at different levels and the number of leaves is abundant (usually more than 5). If the landmark has few details and looks same from different view points (e.g. Arc de Triomphe and Eiffel Tower), the structure of hierarchical summary degenerates to a star (the height of tree is 2) or a line (there is no branches). In such cases, users prefer flat image summary since there is not much gain from the hierarchy representation based on the evaluation criterion introduced in Section VI-D.1. In our dataset of 295 landmarks from 47 countries in 6 continents, the distribution of landmarks is representative of the real world. The general statistics

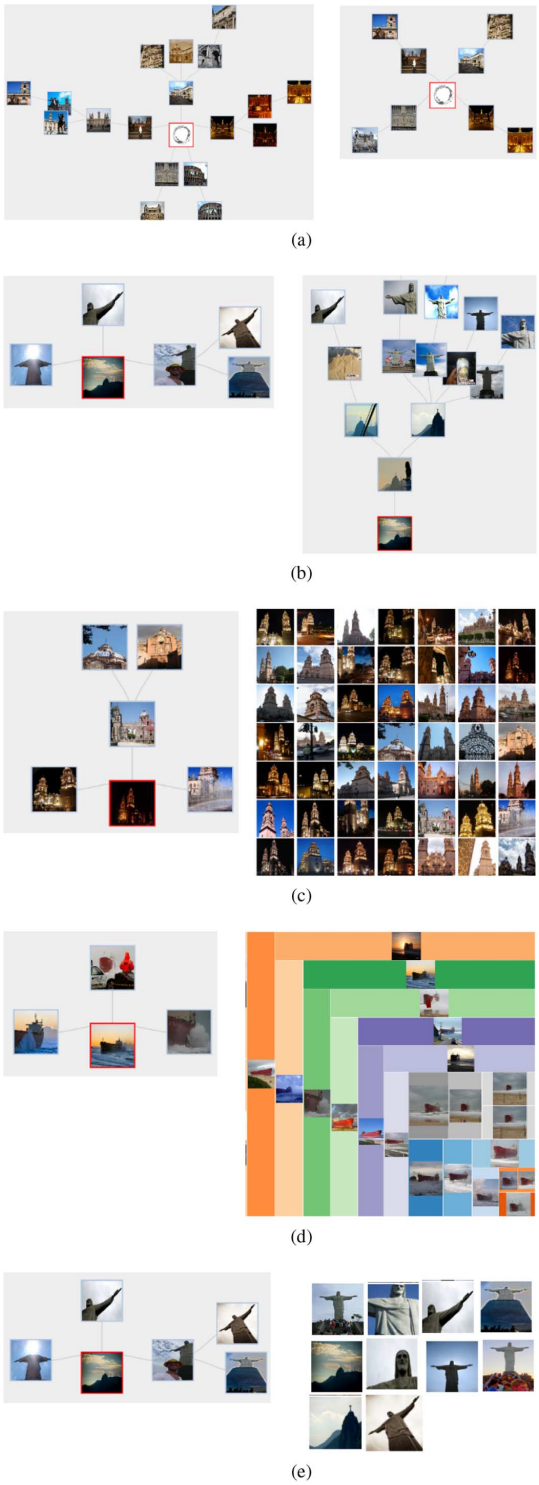


Fig. 11. Case study of five comparisons. Our method is shown on the left and the baselines in comparison is shown on the right. (a) Comparison 1 on L6. (b) Comparison 2 on L2. (c) Comparison 3 on L7. (d) Comparison 4 on L1. (e) Comparison 5 on L7.

on hierarchy structure in Section VI-B and evaluation on properties in Section VI-C justify that most landmarks enjoy the benefit brought by the hierarchical summary.

The summary can be generated offline given an image corpus of the landmark. The major computation time lies in building the match graph in Section V-A. This can be accelerated by paralleling computing to several hours on the total 295 landmarks.

Fig. 12. Study of failed cases in comparison 1 and 3. Our method is shown on the left and the baseline in comparison is shown on the right. (a) Comparison 1 on L10. (b) Comparison 3 on L3.

TABLE IV
DETAILED ANALYSIS OF USER STUDY

comparison	more view points / details	less time / fewer duplicates	prefer the structure
our method	**58**	0	0
B1	30	**8**	0
our method	**15**	**39**	0
B2	13	16	0
our method	0	0	**88**
B3	0	0	61
our method	0	0	**60**
B4	**11**	0	0
our method	0	0	**93**
B5	0	0	55

Steps of detail level estimation in Section V-B, pruning space of trees in Section V-C and searching over pruned space of trees in Section V-D complete within minutes, which can be neglected as compared to the construction of match graph. Furthermore, with more images added to the corpus, we don't need to re-run the entire pipeline. Match graph can be expanded incrementally. In the process, the major space consumption is to store images and features.

An image collection may contain images of nearby landmark. In such case, scale ratio may not correspond to the detail of the landmark. Such issue happens only when the images of nearby landmark and the major landmark are within the same connected component of match graph in Section V-A. This issue involves cleaning images of nearby landmark from the collection. There have been extensive studies on cleaning images of nearby landmark in work [9]–[17]. Our focus is on constructing organized summary from the collection. Cleaning images of nearby landmark can be added as a preprocess step before our solution.

In preprocessing, we set the word number for BoW representation to 20,000 after considering the balance between effectiveness and computation cost. We use approximate kmeans algorithm [12] to cluster visual words. We find that on our dataset

larger $K(K > 20,000)$ does not result in more effective clusters. As for "effective cluster", we mean that a cluster has more than 10 points. It has been shown that large vocabulary results in better discriminative power for matching and retrieval [12]. We try different degree of freedom settings (4,5,6) for pairwise image matching and find that $4dof$ results in best precision. Increasing the degree of freedom brings many false positive matchings.

VII. Conclusion

In this paper, we propose a hierarchical summary to balance users' exploitation and exploration needs on landmark images, which can be used to improve landmark image search result and to provide better representation of the landmark. On this hierarchical summary, exploitation experience is reduced to search operation along a path from root to leaf and exploration experience is reduced to traversal operation on the tree. To achieve this goal, we design two principles for the hierarchical summary. The coherence principle requires that a child image matches and amplifies part of the parent image. The diversity principle requires that different parts of a parent image are matched and amplified by different children. Behind these two principles, we introduce the detail-level concept for a hierarchical summary. Furthermore, we induce three properties of such hierarchical summary: path orderliness property, path smoothness property and branch diversity property. We define exploitation and exploration utility functions for these two types of experience. By combining these two utility functions, we formulate the problem of constructing hierarchical summary as searching over tree space. In experiment, we give general statistics on the hierarchy structure, quantitatively evaluate the three properties of a hierarchical summary and do user study on exploitation and exploration balancing. Results show that the hierarchical summary balances exploitation and exploration experience by providing fewer images with as many details/view points as possible. Compared to flat summary and clustering summary, users think that the hierarchical summary is well organized and it provides better exploitation experience and adds high quality exploration experience.

References

[1] N. Suditu and F. Fleuret, "Iterative relevance feedback with adaptive exploration/exploitation trade-off," in *Proc. 21st ACM Int. Conf. Inf. Knowl. Manage.*, Oct.–Nov. 2012, pp. 1323–1331.

[2] Y. Ke, R. Sukthankar, L. Huston, Y. Ke, and R. Sukthankar, "Efficient near-duplicate detection and sub-image retrieval," in *Proc. ACM Multimedia*, 2004, pp. 869–876.

[3] B. Kulis and K. Grauman, "Kernelized locality-sensitive hashing for scalable image search," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep.–Oct. 2009, pp. 2130–2137.

[4] Y. Jing and S. Baluja, "Pagerank for product image search," in *Proc. 17th Int. Conf. World Wide Web*, Apr. 2008, pp. 307–316.

[5] R. van Zwol *et al.*, "Faceted exploration of image search results," in *Proc. 19th Int. Conf. World Wide Web*, Apr. 2010, pp. 961–970.

[6] R. H. van Leuken, L. G. Pueyo, X. Olivares, and R. van Zwol, "Visual diversification of image search results," in *Proc. 18th Int. Conf. World Wide Web*, Apr. 2009, pp. 341–350.

[7] C. Wu, T. Mei, W. H. Hsu, and Y. Rui, "Learning to personalize trending image search suggestion," in *Proc. 37th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, Jul. 2014, pp. 727–736.

[8] I. Simon, N. Snavely, and S. M. Seitz, "Scene summarization for online image collections," in *Proc. ICCV*, 2007, pp. 1–8.

[9] L. S. Kennedy and M. Naaman, "Generating diverse and representative image search results for landmarks," in *Proc. WWW*, 2008, pp. 297–306.

[10] J. Ye *et al.*, "DLMSearch: Diversified landmark search by photo," in *Proc. ACM Multimedia*, 2012, pp. 905–908.

[11] B. M. Klingner, D. Martin, and J. Roseborough, "Street view motion-from-structure-from-motion," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 953–960.

[12] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2007, pp. 1–8.

[13] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Lost in quantization: Improving particular object retrieval in large scale image databases," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2008, pp. 1–8.

[14] L. Zheng, S. Wang, Z. Liu, and Q. Tian, "Packing and padding: Coupled multi-index for accurate image retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, Jun. 2014, pp. 1947–1954.

[15] Y. Hu, Z. Jin, H. Ren, D. Cai, and X. He, "Iterative multi-view hashing for cross media indexing," in *Proc. ACM Int. Conf. Multimedia*, Nov. 2014, pp. 527–536.

[16] J. Wang *et al.*, "Optimized distances for binary code ranking," in *Proc. ACM Int. Conf. Multimedia*, Nov. 2014, pp. 517–526.

[17] S. Rudinac, A. Hanjalic, and M. Larson, "Generating visual summaries of geographic areas using community-contributed images," *IEEE Trans. Multimedia*, vol. 15, no. 4, pp. 921–932, Jun. 2013.

[18] N. Snavely, S. M. Seitz, and R. Szeliski, "Photo tourism: Exploring photo collections in 3D," in *Proc. SIGGRAPH*, 2006, pp. 835–846.

[19] S. Agarwal, N. Snavely, I. Simon, S. M. Seitz, and R. Szeliski, "Building Rome in a day," in *Proc. IEEE Int. Conf. Comput. Vis.*, Sep.–Oct. 2009, pp. 72–79.

[20] N. Snavely, S. M. Seitz, and R. Szeliski, "Modeling the world from internet photo collections," *Int. J. Comput. Vis.*, vol. 80, no. 2, pp. 189–210, 2008.

[21] Y. S. Avrithis, Y. Kalantidis, G. Tolias, and E. Spyrou, "Retrieving landmark and non-landmark images from community photo collections," in *Proc. ACM Multimedia*, 2010, pp. 153–162.

[22] Q. Hao, R. Cai, Z. Li, L. Zhang, Y. Pang, and F. Wu, "3D visual phrases for landmark recognition," in *Proc. IEEE Conf. on Comput. Vis. Pattern Recog.*, Jun. 2012, pp. 3594–3601.

[23] W. Min, C. Xu, M. Xu, X. Xiao, and B. Bao, "Mobile landmark search with 3D models," *IEEE Trans. Multimedia*, vol. 16, no. 3, pp. 623–636, Apr. 2014.

[24] T. Chen, K. Yap, and D. Zhang, "Discriminative soft bag-of-visual phrase for mobile landmark recognition," *IEEE Trans. Multimedia*, vol. 16, no. 3, pp. 612–622, Apr. 2014.

[25] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *Int. J. Comput. Vis.*, vol. 42, no. 3, pp. 145–175, 2001.

[26] R. H. van Leuken, L. G. Pueyo, X. Olivares, and R. van Zwol, "Visual diversification of image search results," in *Proc. 18th Int. Conf. World Wide Web*, Apr. 2009, pp. 341–350.

[27] L. Li, K. Zhou, G. Xue, H. Zha, and Y. Yu, "Enhancing diversity, coverage and balance for summarization through structure learning," in *Proc. 18th Int. Conf. World Wide Web*, Apr. 2009, pp. 71–80.

[28] X. Li, C. Wu, C. Zach, S. Lazebnik, and J.-M. Frahm, "Modeling and recognition of landmark image collections using iconic scene graphs," in *Proc. ECCV*, 2008, pp. 427–440.

[29] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[30] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[31] O. Chum and J. Matas, "Optimal randomized RANSAC," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 8, pp. 1472–1482, Aug. 2008.

[32] G. H. Golub and C. F. V. Loan, *Matrix Computations*. Baltimore, MD, USA: John Hopkins Univ. Press, 1996.

[33] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, vol. 315, pp. 972–976, 2007.

[34] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *SCIENCE*, vol. 220, no. 4598, pp. 671–680, 1983.

[35] "Oxford building dataset," Vis. Geometry Group, Dept. of Eng. Sci., Univ. of Oxford. Oxford, U.K., 2007 [Online]. Available: http://www.robots.ox.ac.uk/~vgg/data/oxbuildings

[36] Y.-T. Zheng *et al.*, "Tour the world: A technical demonstration of a web-scale landmark recognition engine," in *Proc. MM*, 2009, pp. 961–962.

[37] H. Mannila, "Measures of presortedness and optimal sorting algorithms," *IEEE Trans. Comput.*, vol. C-34, no. 4, p. 318325, Apr. 1985.

[38] R. Lior and O. Maimon, *Data Mining and Knowledge Discovery Handbook*. New York, NY, USA: Springer, 2005.

Jia Chen received the double B.S. degree in mathematics and computer science from Shanghai Jiao Tong University, Shanghai, China, in 2008, and is currently working toward the Ph.D, degree in computer science and engineering at Shanghai JiaoTong University.

His research interests include image annotation, content-based image retrieval, and machine learning.

Qin Jin received the Ph.D. degree in language and information technologies from Carnegie Mellon University (CMU), Pittsburgh, PA, USA, in 2007.

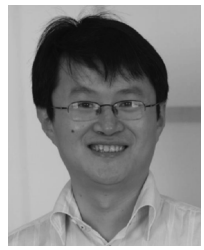
She is currently an Associate Professor with the Computer Science Department, School of Information, Renmin University of China (RUC), Beijing, China, where she also leads the Multimedia Computing Lab. Before joining RUC, she was a a Research Faculty Member with the Language Technologies Institute, CMU, Pittsburgh, PA, USA, from 2007 to 2012, and a Research Scientist with the IBM China Research Laboratory, Beijing, China, in 2012. Her main research interests include speech recognition and understanding, multimedia data analytics, natural language processing, and machine learning in general.

Dr. Jin is a Member of the APSIPA, ISCA, and ACM.

Shenghua Bao received the Ph.D. degree in computer science from Shanghai Jiao Tong University, Shanghai, China, in 2008.

He is currently the Manager of Knowledge Discovery and Analytics Team, IBM Watson Group, San Jose, CA, USA. He was previously Co-Lead of Global Technology Outlook 2014, IBM T.J. Watson Research Center, Yorktown Heights, NY, USA, and a Research Staff Member with the IBM China Research Laboratory, Beijing, China. He has authored or coauthored more than 40 journal and conference papers. He has filed more than 30 U.S. and international patent applications. His research interests include web mining and text analytics.

Dr. Bao is an IBM Master Inventor.



Zhong Su received the Ph.D. degree in computer science from Tsinghua University, Beijing, China, in 2002.

He is currently a Senior Technical Staff Member and Senior Manager of the Cognitive Understanding and Analytics Department, IBM Research China (CRL), Beijing, China. He also currently chairs the Invention and Disclosure Board for CRL. He is an Adjunct Professor with Nankai University, Tianjin, China, and a Guest Professor with the APEX Lab, Shanghai Jiao Tong University, Shanghai, China. He is also the Vice Chairman of Technical Expert Council, IBM Greater China Group. He has authored or coauthored more than 50 papers in top international conferences/journals. He holds more than 40 patents or patents pending.

Dr. Su was the recipient of the IBM Master Inventor Award in 2007 and 2013. His team was the recipient of the Technical Accomplishment Award of IBM Research, as well as the Outstanding Technical Accomplishment Award of IBM Research in 2008, 2010, and 2014.

Shimin Chen received the Ph.D. degree from Carnegie Mellon University, Pittsburgh, PA, USA, in 2005.

He is currently a Professor with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. He previously worked as a Researcher with Intel Laboratories, Pittsburgh, PA, USA, a Senior Researcher with Carnegie Mellon University, and Research Manager with HP Laboratoriess, Beijing, China. His current research interests include data management systems, computer architecture, and big data processing.

Yong Yu received the M.Sc. degree in computer science from East China Normal University, Shanghai, China, in 1986.

He is currently the Ph.D. Candidate Tutor and the Chairman of the E-Generation Technology Research Center, Shanghai Jiao Tong University (SJTU), Shanghai, China. His research interests include semantic web, web mining, information retrieval, and computer vision.

Mr. Yu, as Head Coach of the SJTU ACM-ICPC team, and his team were the recipients of the top prize of the ACM ICPC Championships in 2002, 2005, and 2010.