Tri-Clustered Tensor Completion for Social-Aware Image Tag Refinement

Jinhui Tang, Xiangbo Shu, Guo-Jun Qi, Zechao Li, Meng Wang, Shuicheng Yan, and Ramesh Jain, *Life Fellow, IEEE*

Abstract—Social image tag refinement, which aims to improve tag quality by automatically completing the missing tags and rectifying the noise-corrupted ones, is an essential component for social image search. Conventional approaches mainly focus on exploring the visual and tag information, without considering the user information, which often reveals important hints on the (in)correct tags of social images. Towards this end, we propose a novel tri-clustered tensor completion framework to collaboratively explore these three kinds of information to improve the performance of social image tag refinement. Specifically, the inter-relations among users, images and tags are modeled by a tensor, and the intra-relations between users, images and tags are explored by three regularizations respectively. To address the challenges of the super-sparse and large-scale tensor factorization that demands expensive computing and memory cost, we propose a novel tri-clustering method to divide the tensor into a certain number of sub-tensors by simultaneously clustering users, images and tags into a bunch of tri-clusters. And then we investigate two strategies to complete these sub-tensors by considering (in)dependence between the sub-tensors. Experimental results on a real-world social image database demonstrate the superiority of the proposed method compared with the state-of-the-art methods.

Index Terms—Social image tag refinement, tensor completion, tri-clustering.

1 Introduction

On social websites, users are allowed to upload personal images, label them with freely-chosen tags, and join user groups with common interests. Due to the various professional backgrounds of users, their provided tags tend to be ambiguous, noisy and incomplete. If directly leveraging these noisy and incomplete social tags to perform the tag-based image retrieval, the performance will be far from satisfactory [1]. Therefore, researchers are motivated to develop social image tag refinement approaches [2], [3], [4], [5] to improve the quality of social tags so as to reduce the semantic gap [6], [7]. This task is closely related to tag completion [8], [9], [10], image (re)tagging [11], [12], [13], and image annotation [1], [14]. The goal of social image tag refinement is to automatically complete the missing tags and rectify the noise-corrupted ones.

The prior works related to image tag refinement mainly focus on exploring semantic correlation among tags [3], [15], [16], [17]. For example, Jin *et al.* [15] identified and filtered out the weakly irrelevant annotated tags by exploring tag semantic correlation on WordNet. Xu *et al.* [3] proposed a tag refinement scheme based on tag similarity and relevance by using LDA to mine latent topics. The authors in [18], [13] simultaneously utilized the consistency between image-image and tag-tag intra-relations for tag refinement. In [19], a general subspace learning framework was proposed to explore the visual consistency and the latent structure, and achieved encouraging performance. The common assumption of these approaches is that the visually similar images tend to have the similar semantic tags, and vice versa.

Recently, matrix completion based on low-rank approximation [20] has been explored, which refers to a process of inferring missing entries from a small part of the observed entries in the original matrix between the dyad data (such as word-document in text mining, user-item in recommendation system, and image-feature in image processing). Inspired by matrix completion, several approaches have been proposed in [4], [8], [21], [22] to leverage a small number of observed noisy tags to simultaneously recover the missing tags, remove the noisy tags, and even re-rank the complete tag list [23]. These methods have achieved impressive performance in image tag refinement. However, all the aforementioned methods only explore the visual and tag information, without considering the user information (e.g., user interests and backgrounds) [24] that usually reveals important hints on the (in)correct tags of social images. Therefore, these methods lacking the consideration of

J. Tang, X. Shu and Z. Li are with the School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, 210094, China (e-mail: jinhuitang@njust.edu.cn, shuxb104@gmail.com, zechao.li@njust.edu.cn).

G.-J. Qi is with the Department of Electrical Engineering and Computer Science, University of Central Florida, Orlando, Florida, 32816, USA (email: guojun.qi@ucf.edu).

M. Wang is with the Department of Computer Science, Hefei University of Technology, Anhui, China, (e-mail: eric.mengwang@gmail.com).

S. Yan is with the Department of Electrical and Computer Engineering, National University of Singapore, 117576, Singapore (e-mail: eleyans@nus.edu.sg).

R. Jain is with the University of California, Irvine, CA 92617, USA (e-mail: jain@ics.uci.edu).

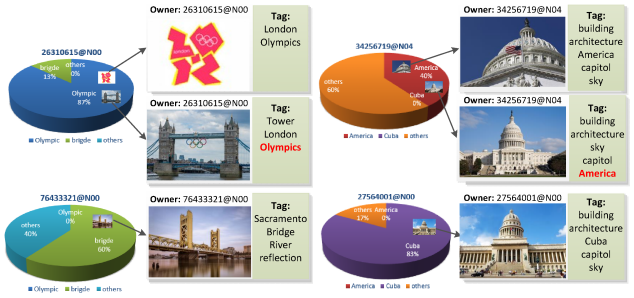


Fig. 1. Exemplary images from Flickr with the associated tags and users. The user information can assist the image tag refinement. The tags (red font) are the completed tags after the refinement.

Fig. 2. Illustration of image groups. Although users generally upload images with multiple interests, the visual difference can obviously cluster them into serval groups.

user information cannot achieve satisfied performance when the visual content and label taxonomy (e.g. WordNet taxonomy) are inconsistent.

1.1 Motivation

We explore the user information to assist social image tag refinement, especially for those images with context information [25], e.g., geo-related tags, event tags, etc. Intuitively, a batch of images uploaded by the same user tend to have close relations, if these images share the same type in terms of events or locations. This hint can help to accurately refine image tags. For example, as illustrated in Figure 1(a), the middle image is more visually similar to the bottom image than the top one. If only considering the visual consistency and tag semantic correlation, the tag "Olympics" cannot be assigned to the middle image. Although this image indeed contains the Olympics logo, it is too small to be captured by the visual information. By introducing the user information, we can find that most of images (accounting for 87% in total) uploaded by the owner of the middle image ("26310615@N00") are related to the "Olympics" event. Thus we can easily infer that the middle image is probably related to the "Olympics" event.

In addition to event tags, the user information can also help to refine geo-tags. For instance, in Figure 1(b), the location information is available in the top image (about U.S. Capitol Building, with geo-tag "America") and the bottom image (about Cuba Capitol Building, with geo-tag "Cuba"), but missing in the middle image. From the visual aspect, the middle image should be more similar to the bottom image than the top one. If we follow the traditional methods which only consider visual consistency and tag semantic correlation, we will incorrectly assign the irrelevant tag "Cuba" to the middle image. However, by analyzing the user information, we find that the owner of the middle image ("34256719@N04") has uploaded many photos with tag "America", but none with tag "Cuba", while the other user ("27564001@N00") has uploaded many photos with tag "Cuba", but never with tag "America". Therefore, the middle image is most likely related to "America" rather than "Cuba".

Although the user information can provide important clues to assist image tag refinement, the misuse of it might lead to errors and noises because users often upload photos with multiple interests. Actually, we can address this issue by leveraging the visual consistency together with the user information. For example, in Figure 2, the images can be clustered into three groups (i.e., the three columns) based on the visual features. The three groups are related to "bridge", "mountain" and "person" respectively. And then the user information can be leveraged to further cluster each group into two events or locations (i.e., the two rows in the figure). For example, the top row of image group 2 is related to the Cascades Mountain, while the bottom row is about Ben Nevis. Therefore, we need to jointly explore the information from users, images and tags to facilitate the image tag refinement.

Recently, some researchers proposed to solve the social image tag refinement problem via tensor completion [26], [27]. They model the inter-relations among users, images and tags via a 3rd-order tensor, and complete an approximate low-rank tensor to refine image tags. The proposed method in [26] refines the tags by directly decomposing the user-image-tag tensor. However, there are several problems in the tensor completion for real-world applications [28]. First, the dimension of the constructed tensor is usually extremely large. The process of tensor completion generates many large-scale temporary matrices and tensors, which requires expensive computing and memory cost. Existing works mainly explore parallel solutions to achieve low complexity and reduce memory cost [29], [30], [28], [31], [32]. Second, the associated 3rd-order tensor is usually very sparse, since the number of observed elements only accounts for a very small ratio compared to the size of the tensor. In order to solve the super sparsity problem of the original tensor, the authors in [27] adopted a ranking optimization scheme to rank tags. However,

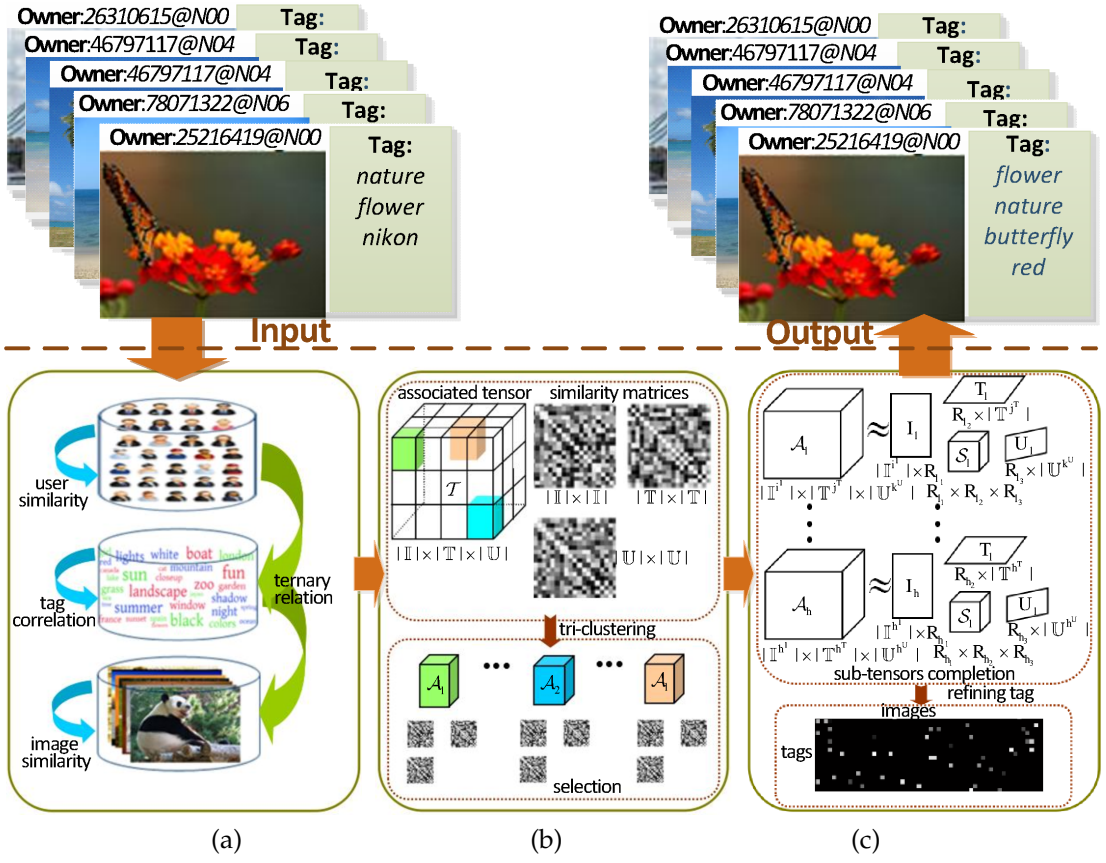


Fig. 3. The framework of Tri-clustered Tensor Completion for social-aware image tag refinement. (a) Relation discovery: find the intra-relations among the homogeneous data; (b) Tri-clustering and sub-tensor selection: divide the original tensor into sub-tensors via tri-clustering, which can simultaneously cluster the users, images and tags, and select the denser sub-tensors; and (c) sub-tensors completion and tag refinement.

it requires several selected "negative" tags before ranking, which will bring in incorrect correlations.

Therefore, to address the above issues, we propose a novel tri-clustered tensor completion (TTC) framework for social image tag refinement, as illustrated in Figure 3. First, we propose an efficient tri-clustering method to divide the original tensor into a certain number of sub-tensors to reduce the computing and memory cost. As to the clustering problem, existing approaches use the associated matrix to model the relationships between two types of data, and then cluster the rows and columns of this matrix simultaneously into co-clusters, which is known as the co-clustering [33] [34]. Motivated by this, the proposed tri-clustering simultaneously identifies the block structures in the rows, columns and tubes. Specifically, it divides the associated tensor into sub-tensors based on the explicit image-tag-user inter-relations and the latent structure of this tensor. Second, to handle the super-sparsity problem of tensor, we select the denser sub-tensors, and then complete these selected sub-tensors. More generally, we consider two variants of the proposed TTC method, i.e., TTC1 and TTC2, based on whether or not the sub-tensors are independent of each other. TTC1 assumes that the sub-tensors are independent, while TTC2 assumes that the sub-tensors are dependent. In experiments, the results of social image tag refinement on a real-world social image database demonstrate the superiority of the proposed method compared with the state-of-the-art methods.

1.2 Contributions

The main contributions of this work can be summarized as follows:

- A novel tri-clustered tensor completion (TTC) framework for social image tag refinement is proposed by solving the low-rank approximation problem of the image-tag-user associated tensor.
- The tri-clustering method is proposed to divide the tensor into several sub-tensors, in order to overcome the challenges of large-scale tensor factorization.
- The sub-tensor completion method is proposed to complete the denser sub-tensors, in order to effectively solve the super-sparse tensor completion problem.
- Two variants of TTC are proposed respectively, by considering two assumptions that whether or not the sub-tensors are independent of each other.

1.3 Organization

The rest of this paper is organized as follows. Section 2 presents the problem definition of the proposed method and briefly introduces the proposed framework. Section 3 discusses how to discover the intra-relations in the data. Section 4 introduces the proposed tri-clustering method and the selection of denser sub-tensors. In Section 5, we details the sub-tensor completion and tag refinement. The optimization procedure is presented in Section 6, followed by the experiments in Section 7. The conclusions and future work are given in the last section.

2 Problem Definition and Framework

In this paper, tensors, matrices, vectors, variables and sets are denoted by calligraphic uppercase letters (e.g., $\mathcal{T}$, $\mathcal{A}$), uppercase letters (e.g., I, T, U), bold lowercase letters (e.g., $\mathbf{d}$), lowercase letters (e.g., x, t, u) and blackboard bold letters (e.g., $\mathbb{I}$, $\mathbb{T}$, $\mathbb{U}$) respectively. Three types of heterogeneous data are collected from photo sharing websites, i.e., the image set $\mathbb{I} = \{x_i\}_{i=1}^{|\mathbb{I}|}$, the tag set $\mathbb{T} = \{t_j\}_{j=1}^{|\mathbb{T}|}$ and the user set $\mathbb{U} = \{u_k\}_{k=1}^{|\mathbb{U}|}$, where $|\mathbb{I}|$, $|\mathbb{T}|$ and $|\mathbb{U}|$ denote the sizes of the image set, the tag set and the user set respectively, and x_i, t_j and u_k denote the *i*-th image, the *j*-th tag and the *k*-th user respectively. $\mathcal{T} \in \mathbb{R}^{|\mathbb{I}|\times|\mathbb{T}|\times|\mathbb{U}|}$ is an image-tag-user associated tensor, where its entry is denoted by $\mathcal{T}_{i,j,k}$ ($1 \leqslant i \leqslant |\mathbb{I}|$, $1 \leqslant j \leqslant |\mathbb{T}|$, and $1 \leqslant k \leqslant |\mathbb{U}|$). If the *i*-th image uploaded by the *k*-th user is annotated with the *j*-th tag, we set $\mathcal{T}_{i,j,k} = 1$, otherwise $\mathcal{T}_{i,j,k} = 0$. In this paper, our goal is to refine these tags by mining the inter- and intra- relations among users, images and tags. Specifically, we can exploit the latent relation of the image-tag-user associated tensor to refine the tags by completing the tensor. Before the tensor completion based on the low-rank approximation, we divide the original tensor into sub-tensors via the proposed tri-clustering method to overcome the challenges of large-scale and super-sparse tensor factorization.

The proposed TTC framework is shown in Figure 3, including three modules. (a) **Relation discovery module.** We construct three similarity matrices (i.e., image-image, user-user and tag-tag similarity matrices) from the data. (b) **Tri-clustering module.** Its goal is to partition the large-scale and super-sparse tensor into a certain number of sub-tensors and select the denser sub-tensors. We expect that the tensor partitioning process can divide the heterogeneous data (images, tags and users) into several groups, in each of which the data is similar in some aspects. After tri-clustering, we select the denser sub-tensors with a relatively larger number of observed entries. (c) **Tensor completion and tag refinement module.** The purpose is to complete the selected sub-tensors to refine social image tags. We employ the tensor Tucker model [35] and low-rank approximation to implement the sub-tensor completion. Here, we investigate the independence or dependence among all the selected sub-tensors, and propose two variants of TTC (i.e., TTC1 and TTC2). TTC1 assumes that these sub-tensors are independent, while TTC2 assumes that these sub-tensors are dependent. Then we integrate these reconstructed sub-tensors into the expected tensor. In order to acquire the image-tag relation matrix, we accumulate the entries along the user axis of this resulted tensor. Finally, we re-rank the tags of images based on the entry values of the obtained image-tag relation matrix.

3 Relation Discovery

Besides the image-tag-user associated tensor $\mathcal{T} \in \mathbb{R}^{|\mathbb{I}|\times|\mathbb{T}|\times|\mathbb{U}|}$ modeling the inter-relations among users, images, and tags, we should also consider the intra-relations in these three types of data.

Let $S^{\mathbb{I}} \in \mathbb{R}^{\mathbb{I}\times\mathbb{I}}$ denote the image similarity matrix, where the similarity between images x_i and x_j is defined as

$$S^{\mathbb{I}}_{i,j} = exp\left(-\frac{||\mathbf{d}_{x_i} - \mathbf{d}_{x_j}||_2^2}{\sigma^2}\right). \tag{1}$$

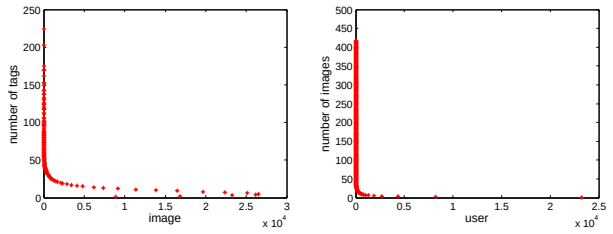
Here $\mathbf{d}_{x_i}$ indicates the low-level features of x_i, and σ is the parameter of the RBF kernel.

The categorical relations and the co-occurrence [36] are two commonly used measures to calculate the similarity of different tags . Similar to the strategy in [27], we compute the tag correlation matrix $S^{\mathbb{T}} \in \mathbb{R}^{|\mathbb{T}|\times|\mathbb{T}|}$ as follows,

$$S^{\mathbb{T}}{}_{i,j} = a_1 \frac{N(t_i,t_j)}{N(t_i)+N(t_j)-N(t_i,t_j)} + (1-a_1)\frac{2\cdot IC(LCS(t_i,t_j))}{IC(t_i)+IC(t_j)}. \tag{2}$$

Here, a_1 denotes the weighted coefficient. $N(t_i)$ is the occurrence count for tag t_i, and $N(t_i,t_j)$ is the co-occurrence count for tags t_i and t_j. Following the same definition of WordNet taxonomy in [37], [38], $LCS(t_i,t_j)$ is the least common subsumer of tags t_i and t_j , and $IC(t_i)$ is the information content of tag t_i. The least common sub-sumer (LCS) of two synsets in WordNet is the sumer that does not have any children that are also the sub-sumer of two synsets. In other words, the LCS of two synsets is the most specific sub-sumer of the two synsets. The first term and the second term of Eq. (2) are the measurements of the co-occurrence and the categorical relation, respectively.

It is observed that users in many social websites with common interest or favor usually have the same or overlapping behaviors. In this paper, it is assumed that two users with higher co-occurrence are more likely to have the common preference and background, and vice versa. The common preference and background between users are reflected by the co-joined groups in social websites. Therefore, we can compute the co-occurrence between users to measure their similarity. The user similarity matrix



(a) Number of tags per image. (b) Number of images per user.

Fig. 4. The statistics on the real-world NUS-WIDE-USER dataset.

$S^{\mathbb{U}} \in \mathbb{R}^{|\mathbb{U}|\times|\mathbb{U}|}$ is defined as follows,

$$S^{\mathbb{U}}_{i,j} = \frac{\text{Num}(u_i, u_j)}{\text{Num}(u_i) + \text{Num}(u_j) - \text{Num}(u_i, u_j)}, \quad (3)$$

where $\text{Num}(u_i, u_j)$ denotes the number of groups which both of users u_i and u_j join, and $\text{Num}(u_i)$ is the number of groups which user u_i joins.

4 Tri-clustering and Sub-tensor Selection

4.1 Tri-Clustering

The image-tag-user associated tensor $\mathcal{T} \in \mathbb{R}^{|\mathbb{I}|\times|\mathbb{T}|\times|\mathbb{U}|}$ constructed from a real-world dataset is very sparse. Taking the NUS-WIDE-USER dataset (see details in Section 7.1) used in the experiments as an example, this dataset contains about $250k$ images, $5k$ tags and $50k$ users. The distributions of the tag numbers per image and the image numbers per owner are shown in Figure 4(a) and Figure 4(b) respectively. In this dataset, the size of the constructed image-tag-user associated tensor is $250k \times 5k \times 50k$ and the number of the non-zero entries is about $2,800k$. That is, the density ratio (number of non-zero entries/tensor size) is $2,800k/(250k\times5k\times50k) \approx 4.5\times10^{-8}$. From the ratio, we can see that this tensor is extremely sparse.

To handle this problem, we propose a novel tri-clustering method to divide the original tensor into several sub-tensors, as shown in Figure 5. The tri-clustering problem can be defined as follows: given a tensor $\mathcal{T} \in \mathbb{R}^{|\mathbb{I}|\times|\mathbb{T}|\times|\mathbb{U}|}$, the goal is to group the $|\mathbb{I}|$ rows, $|\mathbb{T}|$ columns and $|\mathbb{U}|$ tubes of the tensor $\mathcal{T}$ into l^I, l^T and l^U clusters respectively by the corresponding clustering mapping ρ^I, ρ^T and ρ^U simultaneously. And then, we obtain $l^I \times l^T \times l^U$ sub-tensors. To the best of our knowledge, there are only few approaches [39], [40] working on tri-clustering. However, they use two-way co-clustering with two steps rather than an end-to-end solution. They first utilize the inter-relations between two heterogeneous data to implement the co-clustering, and then fuse the two types of co-clustering results to obtain the tri-clustering results. The main advantage of our proposed tri-clustering method is that it is end-to-end. Figure 5 shows the difference between our proposed tri-clustering and the existing approaches.

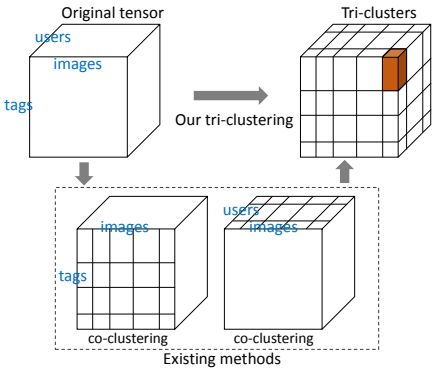


Fig. 5. Visualization for the proposed tri-clustering and the existing methods [39], [40]. The proposed tri-clustering is end-to-end.

Now, we formally define the tri-mapping of tri-clustering by the following formulation:

$$(\rho^I,\rho^T,\rho^U): \mathcal{T}_{i,j,k} \in \mathbb{R}^{|\mathbb{I}|\times|\mathbb{T}|\times|\mathbb{U}|} \rightarrow \hat{\mathcal{T}}_{i^I,j^T,k^U} \in \mathbb{R}^{|\mathbb{I}^{i^I}|\times|\mathbb{T}^{j^T}|\times|\mathbb{U}^{k^U}|}, \quad (4)$$

where $i^I = 1,\cdots,l^I$, $j^T = 1,\cdots,l^T$, $k^U = 1,\cdots,l^U$, $\mathbb{I}^{i^I} \subset \mathbb{I}$, $\mathbb{T}^{j^T} \subset \mathbb{T}$, $\mathbb{U}^{k^U} \subset \mathbb{U}$, and $\hat{\mathcal{T}}_{i^I,j^T,k^U}$ denotes the approximated mean value of the (i^I, j^T, k^U)-th tri-cluster. Here, to measure the quality of tri-clustering, we adopt a squared error loss function. We aim to find the optimal tri-mapping (ρ^I, ρ^T, ρ^U) for the rows, columns, and tubes of the tensor $\mathcal{T}$, so as to minimize the following objective function with respect to the observed value,

$$\min_{(\rho^I,\rho^T,\rho^U)} \sum_{i=1}^{|\mathbb{I}|}\sum_{j=1}^{|\mathbb{T}|}\sum_{k=1}^{|\mathbb{U}|} \mathcal{W}_{i,j,k}(\mathcal{T}_{i,j,k} - \hat{\mathcal{T}}_{\rho^I(i),\rho^T(j),\rho^U(k)})^2, \quad (5)$$

where $\mathcal{W}_{i,j,k}$ is the $0/1$ weight which guarantees that only the observed entries contribute to the loss function. That is, the weights for observed values in $\mathcal{T}$ are set to 1, and others to 0. Then we can obtain the tri-clustering index $i^I = \rho^I(i)$, $j^T = \rho^T(j)$, and $k^U = \rho^U(k)$.

By incorporating the biases of the individual images, tags, users, as well as the tri-cluster mean, we define $\hat{\mathcal{T}}_{i^I,j^T,k^U}$ in Eq. (4) as follow:

$$\hat{\mathcal{T}}_{i^I,j^T,k^U} = \mathcal{T}^{tri}_{i^I,j^T,k^U} + (\mathcal{T}^I_i - \mathcal{T}^I_{i^I}) + (\mathcal{T}^T_j - \mathcal{T}^T_{j^T}) + (\mathcal{T}^U_k - \mathcal{T}^U_{k^U}), \quad (6)$$

where $i^I = \rho^I(i), j^T = \rho^T(j)$, and $k^U = \rho^U(k)$. In Eq. (6), $\mathcal{T}^I_i$, $\mathcal{T}^T_j$ and $\mathcal{T}^U_k$ denote the average values of the i-th lateral slice, the j-th horizontal slice, and the k-th frontal slice of the tensor, respectively,

$$\begin{aligned} \mathcal{T}^I_i &= \frac{1}{|\mathbb{T}| \times |\mathbb{U}|} \sum_{j\in\mathbb{T},k\in\mathbb{U}} \mathcal{T}_{i,j,k}; \\ \mathcal{T}^T_j &= \frac{1}{|\mathbb{I}| \times |\mathbb{U}|} \sum_{i\in\mathbb{I},k\in\mathbb{U}} \mathcal{T}_{i,j,k}; \\ \mathcal{T}^U_k &= \frac{1}{|\mathbb{I}| \times |\mathbb{T}|} \sum_{i\in\mathbb{I},j\in\mathbb{T}} \mathcal{T}_{i,j,k}. \end{aligned} \quad (7)$$

$\mathcal{T}^{tri}_{i^I,j^T,k^U}$, $\mathcal{T}^I_{i^I}$, $\mathcal{T}^T_{j^T}$ and $\mathcal{T}^U_{k^U}$ are defined as follows,

$$\begin{aligned}
&\mathcal{T}^{tri}_{i^I,j^T,k^U}=\frac{1}{|\mathbb{I}^{i^I}|\times|\mathbb{T}^{j^T}|\times|\mathbb{U}^{k^U}|}\sum_{i\in\mathbb{I}^{i^I},j\in\mathbb{T}^{j^T},k\in\mathbb{U}^{k^U}}\mathcal{T}_{i,j,k};\\
&\mathcal{T}^I_{i^I}=\frac{1}{|\mathbb{I}^{i^I}|}\sum_{i\in\mathbb{I}^{i^I}}\mathcal{T}^I_i;\\
&\mathcal{T}^T_{j^T}=\frac{1}{|\mathbb{T}^{j^T}|}\sum_{j\in\mathbb{T}^{j^T}}\mathcal{T}^T_j;\\
&\mathcal{T}^U_{k^U}=\frac{1}{|\mathbb{U}^{k^U}|}\sum_{k\in\mathbb{U}^{k^U}}\mathcal{T}^T_k,
\end{aligned} \tag{8}$$

where $\mathbb{I}^{i^I}$, $\mathbb{T}^{j^T}$ and $\mathbb{U}^{k^U}$ denote the index sets of images, tags, and users in the (i^I, j^T, k^U)-th tri-cluster, respectively. To solve the optimization of the tri-clustering problem in Eq. (5), we adopt an iterative alternating updating algorithm, of which the details will be given in Section 6.1.

4.2 Pruning Strategy

Once the optimal tri-mapping (ρ^I, ρ^T, ρ^U) and tri-clusters $\{(i^I, j^T, k^U)\}^{(l^I,l^T,l^U)}_{(i^I=1,j^T=1,k^U=1)}$ are obtained, the original tensor $\mathcal{T} \in \mathbb{R}^{|\mathbb{I}|\times|\mathbb{T}|\times|\mathbb{U}|}$ can be divided into l ($l = l^I \times l^T \times l^U$) sub-tensors $\mathcal{A}^{i^I,j^T,k^U} \in \mathbb{R}^{|\mathbb{I}^{i^I}|\times|\mathbb{T}^{j^T}|\times|\mathbb{U}^{k^U}|}$, where $i^I = 1, 2, \cdots, l^I$, $j^T = 1, 2, \cdots, l^T$ and $k^U = 1, 2, \cdots, l^U$. Here, many sub-tensors are (nearly) zero. Candès *et al.* [41] proved that the super-sparse matrix cannot be completed when the number of available entries is less than one specific value. Thus, we select the dense sub-tensors for tensor completion in our approach. We measure the density ratio $r(\mathcal{A}^{i^I,j^T,k^U})$ (number of non-zero entries/tensor size) of each sub-tensor and propose a pruning strategy based on a threshold θ. In order to ensure that all images are retained, we reorder the sub-tensors, which share the common index i^I, based on the density ratios $r(\mathcal{A}^{i^I,:,:})$ in descending order. Denote $\{1^{i^I}, 2^{i^I}, \cdots, N^{i^I}\}$ as the descent order of all sub-tensors $A^{i^I,:,:}$, and $\mathcal{A}^{i^I}_n$ as the n^{i^I}-th sub-tensor in $\{1^{i^I}, 2^{i^I}, \cdots, N^{i^I}\}$. Similar to the principal components selection in principal component analysis, we define a formula based on the density contribution rate of the sub-tensors, i.e.,

$$\varphi = \frac{\sum_{n=1}^{M} r(\mathcal{A}^{i^I}_n)}{\sum_{n=1}^{N^{i^I}} r(\mathcal{A}^{i^I}_n)} \geqslant \theta, \tag{9}$$

and select the sub-tensors corresponding to the M largest density ratios from all sub-tensors. This pruning strategy filters out the noise, as well as reduces the computing cost of the following sub-tensor completion step.

5 Tensor Completion and Tag Refinement

In this section, we introduce how to complete the selected sub-tensors to refine the image tags. We first complete each of the selected sub-tensors by tensor decomposition and reconstruction. Then we integrate these reconstructed sub-tensors into the expected tensor. In order to acquire the image-tag relation matrix, we accumulate the entries along the user axis of this resulted tensor. Finally, we re-rank the tags of images based on the entry values of the obtained image-tag relation matrix.

5.1 Sub-Tensor Completion

Suppose we obtain h selected sub-tensors by using the pruning strategy step in Section 4.2. Now the goal is to uncover the missing image-tag-user relations and remove the noise in each selected sub-tensor $\mathcal{A}_i$ ($i = 1, 2, \cdots, h$). The Tucker model [35] seeks a tensor decomposition of $\mathcal{A}_i \in \mathbb{R}^{|\mathbb{I}^{i^I}|\times|\mathbb{T}^{j^T}|\times|\mathbb{U}^{k^U}|}$ as the mode products of a core tensor $\mathcal{S}_i \in \mathbb{R}^{R_1\times R_2\times R_3}$ and $I_i \in \mathbb{R}^{|\mathbb{I}^{i^I}|\times R_1}$, $T_i \in \mathbb{R}^{|\mathbb{T}^{j^T}|\times R_2}$, $U_i \in \mathbb{R}^{|\mathbb{U}^{k^U}|\times R_3}$. The Tucker decomposition provides an extensional solution for tensor completion to estimate the missing entries and remove the noisy ones in an original sub-tensor $\mathcal{A}_i \in \mathbb{R}^{|\mathbb{I}^{i^I}|\times|\mathbb{T}^{j^T}|\times|\mathbb{U}^{k^U}|}$, by reconstructing a low-rank tensor $\tilde{\mathcal{A}}_i \in \mathbb{R}^{|\mathbb{I}^{i^I}|\times|\mathbb{T}^{j^T}|\times|\mathbb{U}^{k^U}|}$ which approximates the original tensor as much as possible,

$$\min_{\mathcal{S}_i,I_i,T_i,U_i} \sum_{i=1}^{h} ||\mathcal{A}_i - \tilde{\mathcal{A}}_i||^2_F, \tag{10}$$

where $\tilde{\mathcal{A}}_i = \mathcal{S}_i \times_1 I_i \times_2 T_i \times_3 U_i$, $\mathcal{S}_i \times_n I_i$ denotes n-mode product of $\mathcal{S}_i$ and I_i, $R_1 < |\mathbb{I}^{i^I}|$, $R_2 < |\mathbb{T}^{j^T}|$ and $R_3 < |\mathbb{U}^{k^U}|$. To avoid overfitting, we also introduce the regularization terms $||I_i||^2_F$, $||T_i||^2_F$ and $||U_i||^2_F$ to the objective function in Eq. (10). Then we obtain

$$\begin{aligned}
\min_{\mathcal{S}_i,I_i,T_i,U_i} \sum_{i=1}^{h} \Big\{ &\tfrac{1}{2} \left\|\mathcal{A}_i - \mathcal{S}_i\times_1 I_i\times_2 T_i\times_3 U_i\right\|^2_F \\
&+\lambda(\|I_i\|^2_F + \|T_i\|^2_F + \|U_i\|^2_F)\Big\},
\end{aligned} \tag{11}$$

where λ is a parameter controlling the weight of the regularization.

However, the objective function in Eq. (11) only considers the inter-relations among the images, tags and users. Actually, in addition to the ternary heterogeneous inter-relations, there are different intra-relations between users, images, and tags, i.e., image similarity matrix $S^{\mathbb{I}} \in \mathbb{R}^{|\mathbb{I}|\times|\mathbb{I}|}$, tag correlation matrix $S^{\mathbb{T}} \in \mathbb{R}^{|\mathbb{T}|\times|\mathbb{T}|}$ and user similarity matrix $S^{\mathbb{U}} \in \mathbb{R}^{|\mathbb{U}|\times|\mathbb{U}|}$. Actually, the optimal matrices I_i, T_i and U_i in Eq. (11) should respectively keep consistent with the corresponding similarity sub-matrices $S^{\mathbb{I}}_i$, $S^{\mathbb{T}}_i$ and $S^{\mathbb{U}}_i$. We use the terms $||I_i I_i^T - S^{\mathbb{I}}_i||^2_F$, $||T_i T_i^T - S^{\mathbb{T}}_i||^2_F$ and $||U_i U_i^T - S^{\mathbb{U}}_i||^2_F$ to measure these consistencies [8]. Then, we introduce these consistency constraints and rewrite the objective function in (11) to formulate the proposed TTC1 model, which assumes that the sub-

tensors are independent with each other, as follows,

$$\min_{\mathcal{S}_i,I_i,T_i,U_i}\sum_{i=1}^{h}\Big\{\tfrac{1}{2}\left\|\mathcal{A}_i-\mathcal{S}_i\times_1 I_i\times_2 T_i\times_3 U_i\right\|_F^2+\lambda(||I_i||_F^2+||T_i||_F^2+||U_i||_F^2)+\beta(||I_iI_i^T-S_i^{\mathbb{I}}||_F^2+||T_iT_i^T-S_i^{\mathbb{T}}||_F^2+||U_iU_i^T-S_i^{\mathbb{U}}||_F^2)\Big\}, \tag{12}$$

where β weights the consistency constraint.

So far, it has been noticed that the model in Eq. (12) is under a more ideal condition that all sub-tensors are independent of each other. In fact, the completed sub-tensor kernels $\mathcal{S}_i$ are related to each other. Therefore, we introduce a public kernel $\mathcal{S}_0$ to bridge the relationship among these sub-tensors. Specifically, we force all the completed sub-tensor kernels $\mathcal{S}_i$ to be close to the public kernel $\mathcal{S}_0$, i.e.,

$$\min\sum_{i=1}^{h}\tau_i||\mathcal{S}_0-\mathcal{S}_i||_F^2+\lambda||\mathcal{S}_i||_F^2, \tag{13}$$

where τ_i $(i=1,2,\cdots,h)$ is the weighted coefficient, which measures the degree of affinity between $\mathcal{S}_0$ and $\mathcal{S}_i$. The greater the value of τ_i is, the more affinity exists between them, and vice versa. In experiments, we set the value of τ_i based on the size of the sub-tensor $\mathcal{A}_i\in\mathbb{R}^{|\mathbb{I}^{i^I}|\times|\mathbb{T}^{j^T}|\times|\mathbb{U}^{k^U}|}$: for $i=1,2,\cdots,h$,

$$\tau_i=\frac{|\mathbb{I}^{i^I}|\times|\mathbb{T}^{j^T}|\times|\mathbb{U}^{k^U}|}{\sum_{i^I,j^T,k^U}^{h^I,h^T,h^U}|\mathbb{I}^{i^I}|\times|\mathbb{T}^{j^T}|\times|\mathbb{U}^{k^U}|}. \tag{14}$$

We integrate the sub-tensor dependency constraint into the objective function of (12), and obtain the objective function g of the proposed TTC2 model as follows,

$$\min_{S_i,I_i,T_i,U_i}g=\min_{S_i,I_i,T_i,U_i}\sum_{i=1}^{h}\Big\{\tfrac{1}{2}\|A_i-S_i\times_1 I_i\times_2 T_i\times_3 U_i\|_F^2+\tau_i||\mathcal{S}_i-\mathcal{S}_0||_F^2+\lambda(||I_i||_F^2+||T_i||_F^2+||U_i||_F^2+||\mathcal{S}_i||_F^2)+\beta(||I_iI_i^T-S_i^{\mathbb{I}}||_F^2+||T_iT_i^T-S_i^{\mathbb{T}}||_F^2+||U_iU_i^T-S_i^{\mathbb{U}}||_F^2)\Big\}. \tag{15}$$

Obviously, the TTC2 model is more in line with the actual situation compared with the TTC1 model. The optimization solution of the objective function g (Eq. (15)) will be introduced in Section 6.2.

5.2 Tag Refinement

When we implement the sub-tensor completion step for a selected sub-tensor $\mathcal{A}_i$, we will obtain the corresponding approximated sub-tensor $\tilde{\mathcal{A}}_i$. We integrate all sub-tensors $\tilde{\mathcal{A}}_i$ $(i=1,\cdots,h)$ as a new tensor $\tilde{\mathcal{T}}\in\mathbb{R}^{|\mathbb{I}|\times|\mathbb{T}|\times|\mathbb{U}|}$ according to the aforementioned sub-tensor indexes.

we then accumulate the entries along the user axis of this new tensor $\tilde{\mathcal{T}}$ to acquire the desired image-tag relation matrix. Specifically, we compute the completed image-tag relation matrix $T^{TI}\in\mathbb{R}^{|\mathbb{T}|\times|\mathbb{I}|}$ by the equation

$$T^{TI}=(\tilde{\mathcal{T}}\times_3\mathbf{1}^T)^T, \tag{16}$$

Algorithm 1 Tri-clustering

Input:
Image-tag-user tensor $\mathcal{T}$, weight tensor $\mathcal{W}$, similarity matrices $S^{\mathbb{I}}$, $S^{\mathbb{T}}$ and $S^{\mathbb{U}}$, parameters l^I, l^T, l^U and p.
Output:
Optimal tri-clustering mapping (ρ^I,ρ^T,ρ^U).
Initialization: Randomly initialize (ρ^I,ρ^T,ρ^U), iter$\leftarrow$1.

1: **repeat**
2: Compute $\hat{\mathcal{T}}_{\rho^I(i),\rho^T(j),\rho^U(k)}$ based on Eq. (6);
3: Update ρ^I: sample one subset $\mathbb{I}'$ from $\mathbb{I}$, and set $\mathbb{I}''=\overline{\mathbb{I}'}\cap\mathbb{I}$, for $\forall i\in\mathbb{I}'$,

$$\rho^I(i)=\arg\min_{1\leqslant i^I\leqslant l^I}\sum_{j\in|\mathbb{T}|}\sum_{k\in|\mathbb{U}|}\mathcal{W}_{i,j,k}(\mathcal{T}_{i,j,k}-\hat{\mathcal{T}}_{\rho^I(i),\rho^T(j),\rho^U(k)})^2;$$

for $\forall i\in\mathbb{I}''$, $\rho^I(i)=\rho^I(i^*)$, where $i^*=\arg\min_{i'\in\mathbb{I}'}S^{\mathbb{I}}_{i,i'}$.
4: Update ρ^T: sample one subset $\mathbb{T}'$ from $\mathbb{T}$, and set $\mathbb{T}''=\overline{\mathbb{T}'}\cap\mathbb{T}$, for $\forall j\in\mathbb{T}'$,

$$\rho^T(j)=\arg\min_{1\leqslant j^T\leqslant l^T}\sum_{i\in|\mathbb{I}|}\sum_{k\in|\mathbb{U}|}\mathcal{W}_{i,j,k}(\mathcal{T}_{i,j,k}-\hat{\mathcal{T}}_{\rho^I(i),\rho^T(j),\rho^U(k)})^2;$$

for $\forall j\in\mathbb{T}''$, $\rho^T(j)=\rho^T(j^*)$, where $j^*=\arg\min_{j'\in\mathbb{T}'}S^{\mathbb{T}}_{j,j'}$.
5: Update ρ^U: sample one subset $\mathbb{U}'$ from $\mathbb{U}$, and set $\mathbb{U}''=\overline{\mathbb{U}'}\cap\mathbb{U}$, for $\forall k\in\mathbb{U}'$,

$$\rho^U(k)=\arg\min_{1\leqslant k^U\leqslant l^U}\sum_{i\in|\mathbb{I}|}\sum_{j\in|\mathbb{T}|}\mathcal{W}_{i,j,k}(\mathcal{T}_{i,j,k}-\hat{\mathcal{T}}_{\rho^I(i),\rho^T(j),\rho^U(k)})^2;$$

for$\forall k\in\mathbb{U}''$,$\rho^U(k)=\rho^U(k^*)$,where $k^*=\arg\min_{k'\in\mathbb{U}'}S^{\mathbb{U}}_{k,k'}$.
6: iter$\leftarrow$iter+1.
7: **until** Convergence.

where $\mathbf{1}\in\mathbb{R}^{|\mathbb{U}|\times1}$ denotes an all-ones vector. Finally, we rerank the completed tags of the image x_i based on the values of $T^{TI}_{:,i}$ in descending order, and select top 10 tags for this image.

6 Optimization Procedure

6.1 Solution for Tri-Clustering

For the optimization problem of the proposed tri-clustering, we initialize the tri-clusters, and then alternately optimize the clustering of the rows (images), columns (tags) and tubes (users) until convergence. However, if we directly implement the iterative alternating updating procedures using the whole data, the computing cost will be considerably large. Therefore, we adopt random sampling, while each of the rest data is assigned to a certain tri-cluster based on the image similarity matrix $S^{\mathbb{I}}$, tag correlation matrix $S^{\mathbb{T}}$ and user similarity matrix $S^{\mathbb{U}}$. In each iteration, the percentage p of the data in the random sampling is set to a pre-defined value. The detailed steps are described in Algorithm 1. The convergence criterion is that the iteration will stop when all the relative costs of the three mapping functions are smaller than a predefined threshold [42].

6.2 Optimization for Sub-Tensor Completion

Taking partial derivatives of the objective function g (Eq. (15)) with respect to $\mathcal{S}_i$, I_i, T_i, U_i and $\mathcal{S}_0$

respectively, we obtain the following equations,

$$\frac{\partial g}{\partial \mathcal{S}_i} = \mathcal{A}_i \times_1 I_i^T \times_2 T_i^T \times_3 U_i^T - \mathcal{S}_i \times_1 (I_i^T I_i) \times_2 (T_i^T T_i) \times_3 (U_i^T U_i) + 2\tau_i(\mathcal{S}_i - \mathcal{S}_0) + 2\lambda \mathcal{S}_i; \tag{17}$$

$$\frac{\partial g}{\partial I_i} = (\mathcal{A}_{i(1)} - I_i G_i^{\mathbb{I}})(G_i^{\mathbb{I}})^T + 2\beta(I_i I_i^T - S_i)I_i + 2\lambda I_i, \tag{18}$$

where $G_i^{\mathbb{I}} = \mathcal{S}_{i(1)}(T_i \otimes U_i)^T$, $T_i \otimes U_i$ denotes the Kronecker product of T_i and U_i, and $\mathcal{A}_{i(n)}$ is the n-mode matricization of $\mathcal{A}_i$;

$$\frac{\partial g}{\partial T_i} = (\mathcal{A}_{i(2)} - I_i G_i^{\mathbb{T}})(G_i^{\mathbb{T}})^T + 2\beta(T_i T_i^T - S_i)T_i + 2\lambda T_i, \tag{19}$$

where $G_i^{\mathbb{T}} = \mathcal{S}_{i(2)}(I_i \otimes U_i)^T$;

$$\frac{\partial g}{\partial U_i} = (\mathcal{A}_{i(3)} - U_i G_i^{\mathbb{U}})(G_i^{\mathbb{U}})^T + 2\beta(U_i U_i^T - S_i)U_i + 2\lambda U_i, \tag{20}$$

where $G_i^{\mathbb{U}} = \mathcal{S}_{i(3)}(I_i \otimes T_i)^T$; and

$$S_0 = \frac{1}{h}\sum\nolimits_{i=1}^{h} \tau_i S_i. \tag{21}$$

Therefore, $\mathcal{S}_i$, I_i, T_i and U_i $(i = 1, \cdots, h)$ in Eq. (15) can be solved by implementing sequentially the following four multiplicative update procedures,

$$\mathcal{S}_i = \mathcal{S}_i \odot \frac{\mathcal{A}_i \times_1 I_i^T \times_2 T_i^T \times_3 U_i^T + 2(\tau_i + \lambda)\mathcal{S}_i}{\mathcal{S}_i \times_1 (I_i^T I_i) \times_2 (T_i^T T_i) \times_3 (U_i^T U_i) + 2\tau_i \mathcal{S}_0}, \tag{22}$$

where $A \odot B$ denotes the elementwise product of matrices A and B;

$$I_i = I_i \odot \frac{\mathcal{A}_{i(1)}(G_i^{\mathbb{I}})^T + 2\beta I_i I_i^T I_i + 2\lambda I_i}{I G_i^{\mathbb{I}}(G_i^{\mathbb{I}})^T + 2\beta S_i^{\mathbb{I}} I_i}; \tag{23}$$

$$T_i = T_i \odot \frac{\mathcal{A}_{i(2)}(G_i^{\mathbb{T}})^T + 2\beta T_i T_i^T T_i + 2\lambda T_i}{T_i G_i^{\mathbb{T}}(G_i^{\mathbb{T}})^T + 2\beta S_i^{\mathbb{T}} T_i}; \tag{24}$$

$$U_i = U_i \odot \frac{\mathcal{A}_{i(3)}(G_i^{\mathbb{U}})^T + 2\beta U_i U_i^T U_i + 2\lambda U_i}{U_i G_i^{\mathbb{U}}(G_i^{\mathbb{U}})^T + 2\beta S_i^{\mathbb{U}} U_i}. \tag{25}$$

In each iteration, when updating one variable, we fix the other variables. The details of the sub-tensor completion algorithm for TTC2 are described in Algorithm 2.

Algorithm 2 Sub-tensor Completion

Input:
Sub-tensor $\mathcal{A}_i$, sub-similarity matrices $S_i^{\mathbb{I}}$, $S_i^{\mathbb{T}}$, $S_i^{\mathbb{U}}$ $(i = 1, \cdots, h)$, ranks R_1, R_2 and R_3, parameters β and λ.
Output:
Optimal low-rank tensor $\tilde{\mathcal{A}}_i$ $(i = 1, \cdots, h)$.
Initialization: Randomly initialize $\mathcal{S}_i$, I_i, T_i and U_i $(i = 1, \cdots, h)$.

```
Compute S_0 with Eq. (21).
for each i ∈ {1, ··· , h} do
    repeat
        Update S_i with Eq. (22).
        Update I_i with Eq. (23).
        Update T_i with Eq. (24).
        Update U_i with Eq. (25).
    until Convergence.
    Update S_0 with Eq. (21).
    return Ã_i = S_i ×_1 I_i ×_2 T_i ×_3 U_i.
end for
```

7 Experiments

7.1 Dataset

To evaluate the effectiveness of the proposed TTC framework, we conduct extensive experiments on a real-world NUS-WIDE-USER dataset, which is extended from the widely-used NUS-WIDE dataset [43]. NUS-WIDE contains 269,648 images with 5,018 unique tags collected from flickr.com, but does not involve the user information that is very crucial in our work. Thus we crawled the user information according to the image IDs provided in NUS-WIDE from flickr.com by using its API. Since some images have bad links, or are deleted by their owners, we only obtain 49,528 user IDs and 247,849 images to construct NUS-WIDE-USER, as shown in Table 1. NUS-WIDE-USER also includes ground-truth of 81 concepts for the images. In experiments, we evaluate the performance of social image tag refinement by F-score $= \frac{2\times\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}}$.

TABLE 1
NUS-WIDE-USER Dataset.

Descriptions	Numbers
Image size	247849
Tag size	5018
User size	49528
Concept size	81
Tags per image	8.47
Images per user	5.0

7.2 Parameter Setting

For tri-clustering, we evaluate the sensitivity of the cluster number, as shown in Figure 6. We can see that the tag refinement performance is insensitive to the cluster number if it is within a suitable range. The proposed method achieves good performance on NUS-WIDE-USER when the value of l^I is about 40, the value of l^T is within [10, 15] and the value of l^U is within [12, 18]. In experiments, we set $l^I = 40$, $l^T = 10$, and $l^U = 12$. The percentage p of the random sampling in each iteration is set to 50%. The radius parameter σ in Eq. (1) is set to 2.5 and the weighted coefficient a_1 in Eq. (2) is set to 0.9 empirically. Figure 7 shows the convergence curve of the optimization in Algorithm 1. It achieves convergence after 20 iterations.

For pruning strategy, since Candès *et al.* in [41] have proved that it seems extremely difficult to complete an approximatively low-rank matrix by matrix completion on the super-sparse matrix, the value of θ is selected according to the practical principle of tensor

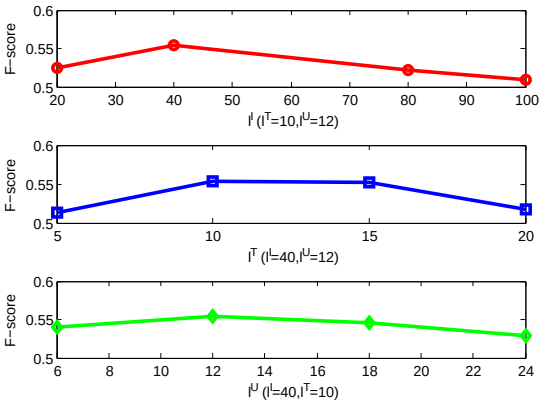


Fig. 6. Performance in terms of F-score by varying the cluster sizes I^I, I^T and I^U.

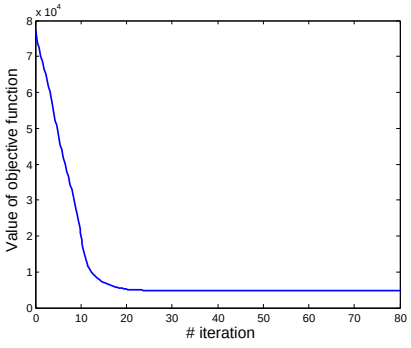


Fig. 7. Convergence curve of the proposed tri-clustering optimization algorithm.

completion. In this work, we set $\theta = 0.80$ and finally obtain $M = 2,814$.

For sub-tensor completion, in order to obtain the optimal weighting parameters β and λ of regularizations, we explore the affection of different parameter settings by employing the grid-search strategy. To balance the convenience and the effectiveness, we explore the optimal parameter settings on a smaller but representative dataset, a subset with 50k images and corresponding users (called NUS-WIDE-USER-50k) of NUS-WIDE-USER. We set $\beta \in \{0.0001, 0.001, 0.0015, 0.01, 0.015, 0.1\}$, $\lambda \in \{0.0001, 0.001, 0.0015, 0.01, 0.015, 0.1\}$ to tune the values of β and λ. Figure 8 shows the impacts of β and λ on the average F-score in a certain sub-tensor. Then, we can achieve the optimal performance for this sub-tensor completion when $\beta = 0.01$ and $\lambda = 0.0015$. The sub-tensor completion needs about $20 \sim 30$ iterations to converge for different sub-tensors.

7.3 Compared Methods

In the experiments, we compare the proposed approach with three image tag refinement methods. The original tags are also employed as the baseline.

- Original Tagging (OT): the original user-contributed tags.

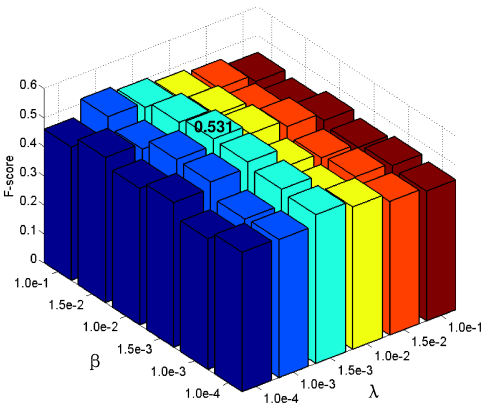


Fig. 8. Parameter tuning results of parameters β and λ.

- Tag Refinement based on Consistency between Visual and Semantic similarity (TRVSC [13]).
- Tag refinement towards Low-Rank, content-tag prior and error sparsity (LR [4]).
- Tag refinement based on Multi-correlation Regularized Tensor Factorization (MRTF [26]).

For the parameter settings of LR and MRTF, we adopt the same way as the TTC, and use the grid-search strategy over the given discrete intervals to obtain the optimal parameters based on the subset NUS-WIDE-USER-50k. For TRVSC, the values of model parameters can be found in [13].

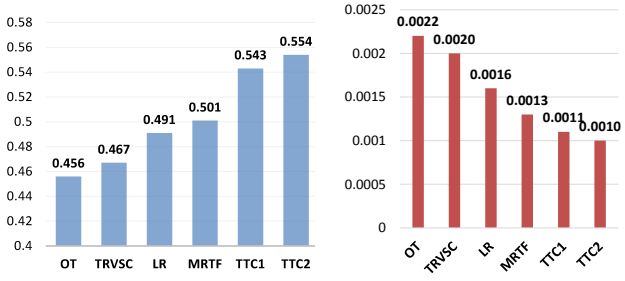
7.4 Comparison and Analysis

All approaches in the experiments are executed using MATLAB on a DELL Server with a 12-core 2.67 GHz CPU and 32 GB memory. The convergence time of tri-clustering is about 5.0 hours. Figure 9 shows some examples of the generated tri-clusters, i.e., sub-tensors, where the data surrounded by the same solid-line box belongs to the same group. All the images, tags and users are clustered into several groups. It is noted that: 1) the user, image and tag information are equally important in the tri-clustering process; 2) the group results are decided by all the three kinds of information (i.e., image, tag and user); 3) the image, tag and user data within the same group are extremely close. For example, in the first group, all the images not only have the similar visual contents, but also have the common users (e.g., "34134691@N00", "64725810@N00") and the common tags (e.g., "castle" and "Germany"); in the second group, the images have some common users (e.g., "7761395@N02", "14645458@N00", and "20791254@N00") and some common tags (e.g., "California" and "aerial"), while some of the images are visually similar (the third image and the sixth one).

Figure 10(a) shows the average F-scores obtained by different image tag refinement methods[1]. We find

1. LR and TRVSC is performed on NUS-WIDE-USER without the user data since these methods do not consider the user information.

Fig. 9. Visualizations of the obtained tri-clusters by the tri-clustering method. The data surrounded by the same solid line box belong to the same sub-tensor. Here, we list no more than 10 tags for each image and no more than 9 images for each tri-cluster due to the space limitation.



(a) F-socre. (b) False positive rate.

Fig. 10. Average performance of different methods for tag refinement.

that besides the original tagging, TRVSC has the worst performance, since it does not consider the distinctive characteristics of the dataset, i.e., low-rank and error sparsity. That is why LR is more superior compared to TRVSC on the sparse dataset. MRTF and TTC utilize the inter- and intra- relations among images, tags and users, while LR only leverages the relations between images and tags. Thus MRTF and TTC (including TTC1 and TTC2) perform better than LR, which shows the advantage of incorporating the user information. Although the tensor models inter-relations among the images, tags and users, it also brings in a new challenge, i.e., the more severe sparsity. As a result, MRTF's performance degrades since it cannot well address this super-sparsity problem. Fortunately, with the previous processing of the tri-clustering, TTC can well address the sparsity problem of the 3-rd order tensor. Thus it significantly outperforms all the other methods. Moreover, the F-score obtained by TTC2 is 0.554, which is about one percent higher than 0.543 of TTC1, since TTC2 considers the dependency between the sub-tensors while TTC1 does not. Besides, we show the false positive rates of all the approaches in Figure 10(b). We can see that TTC (including TTC1 and TTC2) achieves lower false positive rate than the other methods. Statistically, the average number of the user-provided tags per image on the test set is 8.47, while it becomes 10 after applying the proposed method. After tag refinement, the average number of added tags per image is 3.56, while the average number of deleted tags per images is 2.03.

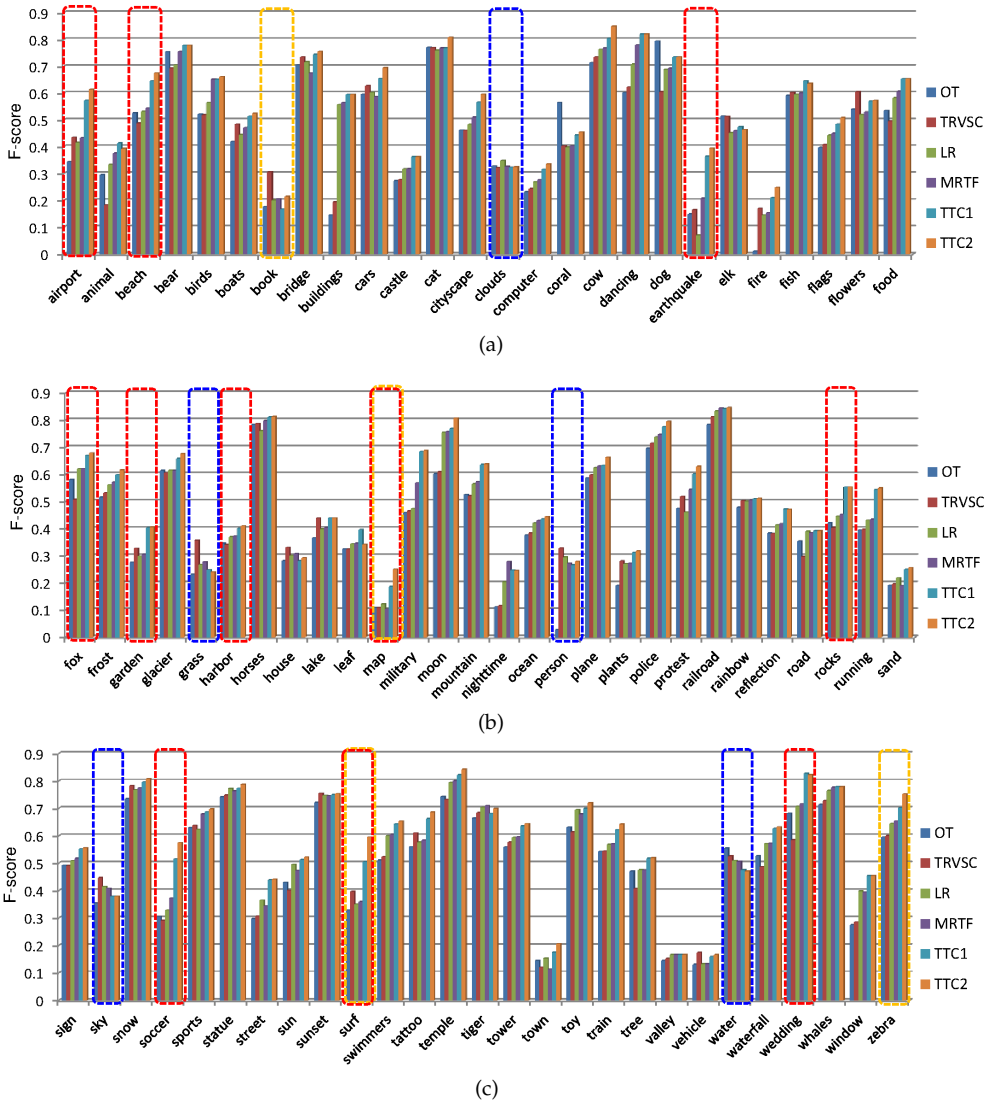


Fig. 11. Detailed F-scores on the 81 concepts by different methods. For some special geo-related tags (e.g., "airport", "beach", "garden" and "harbor"), event tags (e.g., "earthquake", "soccer", "surf" and "wedding") and ambiguous tags (e.g., "fox", "map" and "rocks"), which are denoted with red dotted line boxes, the proposed methods show remarkable improvements. Moreover, TTC2 outperforms TTC1 for the very rare tags (e.g., "book", "map", "soccer", "surf" and "zebra"), which are denoted with yellow dotted line boxes.

Then we analyze the F-scores obtained by different methods on all the 81 concepts and show the results in Figure 11. Compared with OT, we find that all the tag refinement methods improve the quality of almost all the 81 tags. By utilizing the user information, MRTF and TTC achieve better performance than LR and TRVSC, especially for those summarized or complex tags (e.g., "dancing", "military", "nighttime", "cityscape" and "protest"). Besides, for some special geo-related tags (e.g., "airport", "beach", "garden" and "harbor"), event tags (e.g., "earthquake", "soccer", "surf" and "wedding") and ambiguous tags (e.g., "fox", "map" and "rocks"), which are denoted with red dotted line boxes in Figure 11, the proposed methods show remarkable improvements compared to the other methods, since they can solve the super-sparsity problem and well uncover the latent relationships between images and tags.

Since the frequently-used tags (e.g., "clouds", "grass", "person", "sky" and "water"), as enclosed by blue dotted line boxes in Figure 11, are not very sparse, the obtained F-scores on them do not achieve significant improvements by using the proposed methods. For most tags, the F-scores obtained by TTC are higher than or equal to the ones obtained by the other methods. The performances of TTC1 and TTC2 are very close for most of the tags except for some relatively special tags (e.g., "book", "map", "soccer", "surf" and "zebra"), which are denoted with yellow dotted line boxes. After observing the number of relevant images for the 81 concepts, we notice that these tags are rarer in images. Therefore, it is concluded that TTC2 is more robust than TTC1 especially for the very rare tags.

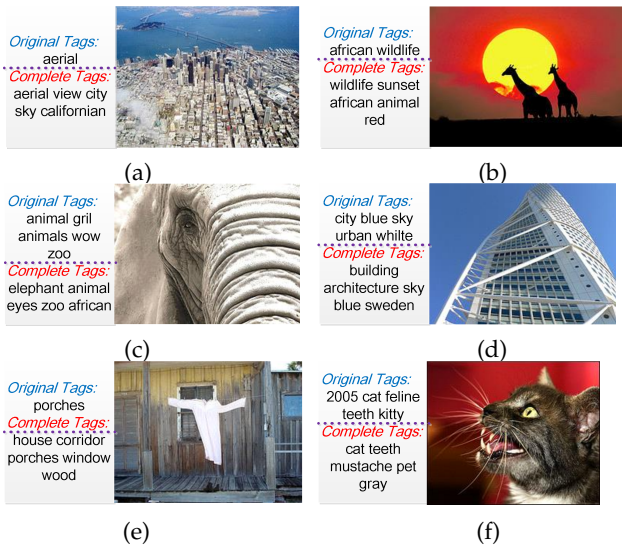
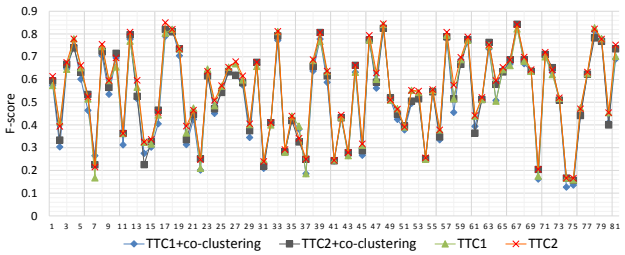


Fig. 12. Exemplary tag refinement results by the proposed framework. For each image, the top 5 tags are shown.

We also present some image tag refinement results of TTC2 for some specific cases to demonstrate the effectiveness of the proposed framework. For each image, the top 5 refined tags are shown in Figure 12. Our method is effective for the tags that are hard to infer by only using the visual and tag information, but can be discovered by utilizing the user information. For example, as shown in Figure 12(a), the Golden Gate Bridge is a landmark of California, thus this image should be tagged with the geo-tags "Californian" or "San Francisco". It is very difficult to restore the relation between the geo-tag "Californian" and the image by only mining the visual and tag information, but the proposed methods can do it well based on the user background information. Similarly in Figure 12(d), since the Malmo Building is regarded as one of the most famous symbols in Sweden, it is reasonable to assign the tag "Sweden" to this image. After refinement, the tag "African" is added to the image in Figure 12(c). Although we can hardly infer the relation between the image and "African" from either its visual content or tag semantic relations, one credible explanation is that the user who uploaded this image also uploaded other images with the tag "African" by mining the user information.

Besides completing the missing tags, the proposed TTC method can also remove the noisy tags. Examples in Figure 12(c) and (f) denote the cases with original noisy tags, while Figure 12(a) and (d) show the cases with original incomplete and missing tags. We can also see that TTC can significantly improve the quality of the tags for locally abstract images, e.g., Figure 12(c) and (d), and globally complex images like Figure 12(a). In other words, besides improving the tag quality for common images, the proposed method can also be used to refine the tags of the locally abstract images and the globally complex images with



(a)

Methods	TTC1+co-clustering	TTC2+co-clustering	TTC1	TTC2
F-score	0.522	0.535	0.543	**0.554**

(b)

Fig. 13. The performance comparisons of co-clustering and the proposed tri-clustering. (a) and (b) are the individual F-scores and average F-score respectively. In (a), horizontal axis denotes the 81 concepts.

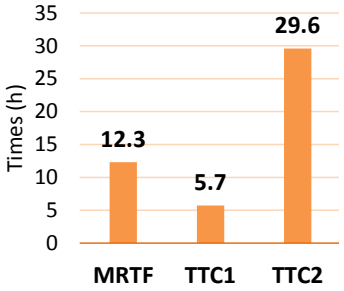


Fig. 14. Comparison of the computing times (h) on NUS-WIDE-USER.

the context information and semantic relations.

To illustrate the superiority of the proposed tri-clustering method, we compare it to the co-clustering methods. To adapt co-clustering into the proposed TTC framework, we extend the co-clustering from 2 dimensions to 3 dimensions by changing the definition of tri-cluster mean in Eq. (6), namely $\hat{\mathcal{T}}_{i^I,j^T,k^U} = \mathcal{T}^{tri}_{i^I,j^T,k^U}$. For this extended co-clustering in TTC1 and TTC2, we name them as TTC1+co-clustering and TTC2+co-clustering, respectively. The individual and average F-scores are plotted in Figure 13. For almost all the 81 concepts, TTC1 and TTC2 perform better than TTC1+co-clustering and TTC2+co-clustering, respectively. Therefore, the proposed tri-clustering can improve the tag refinement performance compared to the extended co-clustering method. Moreover, the running time of the proposed tri-clustering is about 5.0 hours, which is less than the 9.8 hours of co-clustering on NUS-WIDE-USER.

We also compare TTC with MRTF in terms of the computing time and present the results in Figure 14. We can see that TTC1 executes much faster than MRTF, while TTC2 executes a bit slower than MRTF. Thus we can conclude that the proposed TTC framework performs better than MRTF in terms of both effectiveness and efficiency, while assuming the independence of sub-tensors. Actually, for practical applications,

image tag refinement is a pre-processing step of tag-based image retrieval, and is usually implemented in an off-line way, in which the effectiveness is much more important than the efficiency. In this scenario, considering the dependence of sub-tensors in the proposed framework is more practical and applicable to promote the refinement accuracy.

8 Conclusions and Future Work

In this paper, we have proposed a novel tri-clustered tensor completion (TTC) framework, which has collaboratively explored the user information, visual consistency and tag semantic correlation, to improve the performance of social image tag refinement. In this framework, the tri-clustering and sub-tensor completion methods have been proposed in order to address the challenges of large-scale and super-sparse tensor completion. In particular, it divides the original image-tag-user associated tensor into sub-tensors by the proposed tri-clustering method. And then we investigate two strategies to complete these sub-tensors by considering (in)dependence between the sub-tensors. Experimental results on a real-world community-contributed database have demonstrated the superiority of the proposed framework compared with the state-of-the-art methods. In the future, we plan to make the proposed framework applicable to social-aware video tag refinement.

Acknowledgments

This work was partially supported by the 973 Program of China (Project No. 2014CB347600), the National Natural Science Foundation of China (Grant No. 61522203 and 61402228), and the National Ten Thousand Talent Program of China (Young Top-Notch Talent).

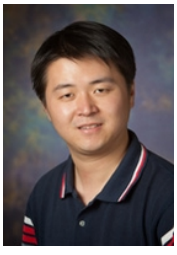
References

[1] J. Z. Wang, D. Geman, J. Luo, and R. M. Gray, "Real-world image annotation and retrieval: An introduction to the special section," *IEEE TPAMI*, vol. 30, no. 11, pp. 1873–1876, 2008.
[2] C. Wang, F. Jing, L. Zhang, and H.-J. Zhang, "Content-based image annotation refinement," in *CVPR*, 2007.
[3] H. Xu, J. Wang, X.-S. Hua, and S. Li, "Tag refinement by regularized LDA," in *ACM Multimedia*, 2009.
[4] G. Zhu, S. Yan, and Y. Ma, "Image tag refinement towards low-rank, content-tag prior and error sparsity," in *ACM Multimedia*, 2010.
[5] X. Li, C. G. Snoek, and M. Worring, "Learning social tag relevance by neighbor voting," *IEEE TMM*, vol. 11, no. 7, pp. 1310–1322, 2009.
[6] G.-J. Qi, C. Aggarwal, J. Han, and T. Huang, "Mining collective intelligence in diverse groups," in *WWW*, 2012.
[7] Z.-H. Deng, H. Yu, and Y. Yang, "Image tagging via cross-modal semantic mapping," in *ACM Multimedia*, 2015.
[8] L. Wu, R. Jin, and A. K. Jain, "Tag completion for image retrieval," *IEEE TPAMI*, vol. 35, no. 3, pp. 716–727, 2013.
[9] Z. Lin, G. Ding, M. Hu, J. Wang, and X. Ye, "Image tag completion via image-specific and tag-specific linear sparse reconstructions," in *CVPR*, 2013.
[10] X. Li, Y.-J. Zhang, B. Shen, and B.-D. Liu, "Image tag completion by low-rank factorization with dual reconstruction structure preserved," in *ICIP*, 2014.
[11] L. Chen, D. Xu, I. W. Tsang, and J. Luo, "Tag-based web photo retrieval improved by batch mode re-tagging," in *CVPR*, 2010.
[12] N. Zhou, W. K. Cheung, G. Qiu, and X. Xue, "A hybrid probabilistic model for unified collaborative and content-based image tagging," *IEEE TPAMI*, vol. 33, no. 7, pp. 1281–1294, 2011.
[13] D. Liu, X.-S. Hua, M. Wang, and H.-J. Zhang, "Image retagging," in *ACM Multimedia*, 2010.
[14] J. Tang, R. Hong, S. Yan, T.-S. Chua, G.-J. Qi, and R. Jain, "Image annotation by k nn-sparse graph-based label propagation over noisily tagged web images," *ACM TIST*, vol. 2, no. 2, p. 14, 2011.
[15] Y. Jin, L. Khan, L. Wang, and M. Awad, "Image annotations by combining multiple evidence & wordnet," in *ACM Multimedia*, 2005.
[16] C. Wang, F. Jing, L. Zhang, and H.-J. Zhang, "Image annotation refinement using random walk with restarts," in *ACM Multimedia*, 2006.
[17] D. Liu, X.-S. Hua, L. Yang, M. Wang, and H.-J. Zhang, "Tag ranking," in *WWW*, 2009.
[18] J. Liu, B. Wang, M. Li, Z. Li, W. Ma, H. Lu, and S. Ma, "Dual cross-media relevance model for image annotation," in *ACM Multimedia*, 2007.
[19] Z. Li, J. Liu, J. Tang, and H. Lu, "Robust structured subspace learning for data representation," *IEEE TPAMI*, vol. 37, no. 10, pp. 2085–2098, 2015.
[20] E. J. Candes and Y. Plan, "Matrix completion with noise," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 925–936, 2010.
[21] Z. Li, J. Liu, X. Zhu, T. Liu, and H. Lu, "Image annotation using multi-correlation probabilistic matrix factorization," in *ACM Multimedia*, 2010.
[22] X. Liu, S. Yan, T.-S. Chua, and H. Jin, "Image label completion by pursuing contextual decomposability," *ACM TOMM*, vol. 8, no. 2, p. 21, 2012.
[23] J. Zhuang and S. C. Hoi, "A two-view learning approach for image tag ranking," in *WSDM*, 2011.
[24] P. Cui, S.-W. Liu, W.-W. Zhu, H.-B. Luan, T.-S. Chua, and S.-Q. Yang, "Social-sensed image search," *ACM Transactions on Information Systems*, vol. 32, no. 2, p. 8, 2014.
[25] G.-J. Qi, C. C. Aggarwal, Q. Tian, H. Ji, and T. S. Huang, "Exploring context and content links in social media: A latent space method," *IEEE TPAMI*, vol. 34, no. 5, pp. 850–862, 2012.
[26] J. Sang, J. Liu, and C. Xu, "Exploiting user information for image tag refinement," in *ACM Multimedia*, 2011.
[27] J. Sang, C. Xu, and J. Liu, "User-aware image tag refinement via ternary semantic analysis," *IEEE TMM*, vol. 14, no. 3, pp. 883–895, 2012.
[28] P. E. Crandall and M. J. Quinn, "Block data decomposition for data-parallel programming on a heterogeneous workstation network," in *Proceedings the 2nd International Symposium on High Performance Distributed Computing*, 1993.
[29] R. M. Czekster, C. A. De Rose, P. Fernandes, A. M. de Lima, and T. Webber, "Kronecker descriptor partitioning for parallel algorithms," in *Proceedings of the Spring Simulation Multiconference*, 2010.
[30] A. Benoit, B. Plateau, and W. J. Stewart, "Memory-efficient kronecker algorithms with applications to the modelling of parallel systems," *Future Generation Computer Systems*, vol. 22, no. 7, pp. 838–847, 2006.
[31] J. Tang, Z. Li, M. Wang, and R. Zhao, "Neighborhood discriminant hashing for large-scale image retrieval," *IEEE IIP*, vol. 24, no. 9, pp. 2827–2840, 2015.
[32] E. Papalexakis, C. Faloutsos, and N. Sidiropoulos, "Parcube: Sparse parallelizable tensor decompositions," *Machine Learning and Knowledge Discovery in Databases*, pp. 521–536, 2012.
[33] I. S. Dhillon, "Co-clustering documents and words using bipartite spectral graph partitioning," in *KDD*, 2001.
[34] X. He, D. Cai, H. Liu, and J. Han, "Image clustering with tensor representation," in *ACM Multimedia*, 2005.
[35] L. R. Tucker, "Some mathematical notes on three-mode factor analysis," *Psychometrika*, vol. 31, no. 3, pp. 279–311, 1966.
[36] L. Wu, X.-S. Hua, N. Yu, W.-Y. Ma, and S. Li, "Flickr distance: a relationship measure for visual concepts," *IEEE TPAMI*, vol. 34, no. 5, pp. 863–875, 2012.
[37] P. Resnik, "Using information content to evaluate semantic similarity in a taxonomy," *arXiv*, 1995.

[38] D. Lin, "Using syntactic dependency as local context to resolve word sense ambiguity," in *ACL*, 1997.
[39] L. Zhao and M. J. Zaki, "Tricluster: an effective algorithm for mining coherent clusters in 3d microarray data," in *ACM SIGMOD*, 2005.
[40] Q. Zhou, G. Xu, and Y. Zong, "Web co-clustering of usage network using tensor decomposition," in *International Joint Conferences on Web Intelligence and Intelligent Agent Technologies*.
[41] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Foundations of Computational mathematics*, vol. 9, no. 6, pp. 717–772, 2009.
[42] Z. Li, J. Liu, Y. Yang, X. Zhou, and H. Lu, "Clustering-guided sparse structural learning for unsupervised feature selection," *IEEE TKDE*, vol. 26, no. 9, pp. 2138–2150, 2014.
[43] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng, "Nus-wide: a real-world web image database from national university of singapore," in *CIVR*, 2009.

Jinhui Tang is a Professor in School of Computer Science and Engineering, Nanjing University of Science and Technology, China He received his B.E. and Ph.D. degrees from the University of Science and Technology of China in 2003 and 2008 respectively. His current research interests include multimedia search and computer vision. He is a co-recipient of the Best Student Paper Award in MMM 2016, and Best Paper Awards in ACM MM 2007, PCM 2011 and ICIMCS 2011.

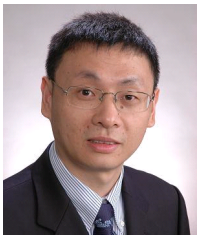
Xiangbo Shu is currently a PhD candidate of School of Computer Science and Engineering, Nanjing University of Science and Technology, China. From 2014 to 2015, he was also an visiting scholar in the Department of Electrical and Computer Engineering at National University of Singapore. Her research interests include social multimedia computing and computer vision. He received the Best Student Paper Award in MMM 2016 and the Best Paper Finalist in ACM MM 2015.



Guo-Jun Qi is an Assistant Processor in the Department of Electrical Engineering and Computer Science, University of Central Florida. He received the B.S. degree from University of Science and Technology of China in 2005. He received his PhD in 2013 from the Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign. His research interests include machine learning, computer vision, and multimedia. He received the Best Paper Award from ACM MM 2007.

Zechao Li is currently an Associate Professor in Nanjing University of Science and Technology, China. He received the Ph.D degree from National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences in 2013. He received the B.E. degree from University of Science and Technology of China in 2008. His research interests include multimedia, machine learning, etc. He received the 2013 President Scholarship of Chinese Academy of Science.

Meng Wang is a Professor in the Hefei University of Technology, China. He received the B.E. degree and Ph.D. degree in the Special Class for the Gifted Young and the Department of Electronic Engineering and Information Science from the University of Science and Technology of China, China, respectively. His current research interests include multimedia computing. He received the best paper awards from ACM MM 2009 and ACM MM 2010, the best paper award from MMM 2010, the best paper award from ICIMCS 2014, and the best demo award from ACM MM 2012.



Shuicheng Yan is an Associate Professor at the Department of Electrical and Computer Engineering at National University of Singapore. Dr. Yan's research areas include machine learning, computer vision and multimedia, and he has authored/co-authored hundreds of technical papers over a wide range of research topics, with Google Scholar citation $>$15,000 times and H-index 52. He has been serving as an associate editor of IEEE TCSVT and ACM TIST. He received the Best Paper Awards from ACM MM'13 (Best Paper and Best Student Paper), ACM MM 2012 (Best Demo), PCM 2011, ACM MM 2010, ICME 2010, MMM 2016 and ICIMCS 2009, the winner prizes of the classification task in PASCAL VOC 2010-2012, 2011 Singapore Young Scientist Award, and 2012 NUS Young Researcher Award.

Ramesh Jain is the Bren Professor of Information and Computer Science, Department of Computer Science, University of California, Irvine. He has been an active researcher in multimedia information systems, image databases, machine vision, and intelligent systems, with H-index 79 in Google Scholar. His current research is on experiential systems and their applications. Dr. Jain was the founding Editor-in-Chief of IEEE Multimedia Magazine and Machine Vision and Applications and serves on the editorial boards of several magazines in multimedia, business, and image and vision processing. Dr. Jain is a Fellow of IEEE, ACM, IAPR, AAAI and SPIE.