

A Hybrid Probabilistic Model for Unified Collaborative and Content-based Image Tagging

Ning Zhou, William K. Cheung, Guoping Qiu, and Xiangyang Xue*

Abstract—Increasing availability of large quantities of user contributed images with labels has provided opportunities to develop automatic tools to tag images to facilitate image search and retrieval. In this paper, we present a novel hybrid probabilistic model (HPM) which integrates low-level image features and high-level user provided tags to automatically tag images. For images without any tags, HPM predicts new tags based solely on the low-level image features. For images with user provided tags, HPM jointly exploits both the image features and the tags in a unified probabilistic framework to recommend additional tags to label the images. The HPM framework makes use of the tag-image association matrix (TIAM). However, since the number of images is usually very large and user provided tags are diverse which means that TIAM is very sparse thus making it difficult to reliably estimate tag to tag co-occurrence probabilities. We developed a collaborative filtering method based on non-negative matrix factorization (NMF) for tackling this data sparsity issue. Also, an L_1 norm kernel method is used to estimate the correlations between image features and semantic concepts. The effectiveness of the proposed approach has been evaluated using three databases containing 5,000 images with 371 tags, 31,695 images with 5,587 tags, and 269,648 images with 5,018 tags, respectively.

Index Terms—Automatic Image Tagging, Collaborative Filtering, Feature Integration, Non-negative Matrix Factorization, Kernel Density Estimation

1 INTRODUCTION

RECENT proliferation of user contributed digital photos on the Internet¹ has created the need for tools to help users to quickly and accurately locate the correct photos they are looking for. In the past decade, image retrieval has attracted attentions from researchers in various sub-fields in computer science including artificial intelligence, computer vision, database, etc. Content-based image retrieval (CBIR) [12], [19], [49], [53], where computer vision and image processing techniques are used to extract low-level visual features as metadata so that image retrieval can be performed by directly comparing the color, texture, object shape as well as other visual cues, has emerged as a very active research area. Although CBIR has some initial success and has triggered some early excitement in the respective research community, the effectiveness of putting this approach into practice is still unclear, partly due to the semantic gap problem [49].

One way to reduce the semantic gap is by introducing high-level knowledge through user labeling (also called tagging). Carefully chosen tags can improve image retrieval accuracy, but the tagging process is known to be tedious and labor-intensive [41]. What being desired is a method which can automatically or semi-automatically annotate images. Recently, methods aim to automatically produce a set of semantic labels for images based on their visual contents have attracted a lot of attention [6], [8], [9], [13], [16], [17], [18], [29], [30], [31], [32], [36], [37], [39], [54], [59]. These methods first extract low-level features of images and then build a mathematical model to associate low-level image contents with annotations. We refer to such methods as *content-based image tagging* (CBIT). As these methods are still purely content-based, the extent to which they can fill the semantic gap is intrinsically limited.

With the recent development of Web 2.0, many content sharing platforms, e.g., Flickr [20], PhotoStuff [26], among others, have provided annotation functions to enable users to tag images at anytime from anywhere. Therefore, there already exist a large number of images tagged with words describing the image contents perceived by human users. These tagged image databases contain rich information about the semantic meanings of the images, which can in turn be exploited to automatically tag the untagged images or add extra tags to images with a few existing tags. One way to exploit those existing user provided tags so as to automatically tag a given image is to simply rank the related tags based on tag co-occurrence in the whole data set [48]. The idea is similar to that of collaborative filtering (CF) [23]. Given an image with one or more user provided

- N. Zhou was with the School of Computer Science, Fudan University, Shanghai 201203, China. He is now with the Department of Computer Science, University of North Carolina, Charlotte, NC 28223, USA. E-mail: nzhou@unc.edu.
- W.K. Cheung is with the Department of Computer Science, Hong Kong Baptist University, Kowloon Tong, Hong Kong. E-mail: william@comp.hkbu.edu.hk.
- G. Qiu is with the School of Computer Science, The University of Nottingham, Nottingham NG8 1BB, UK. E-mail: qiu@cs.nott.ac.uk.
- X. Xue is with the School of Computer Science, Fudan University, Shanghai 201203, China. E-mail: xyxue@fudan.edu.cn.

* Corresponding author: Dr. Xiangyang Xue

1. According to <http://www.techcrunch.com/2008/11/03/three-billion-photos-at-flickr/>, there have been more than 3 billion photos uploaded to the photo sharing site Flickr [20] as of November 2008.

tags, we can predict additional tags for the image by exploiting the correlations between tags. For instance, with two groups of images— one tagged with “sky” and “tree” and the other tagged with “tree” and “grass”, a new image tagged with only “grass” could be predicted to have the tag “sky” even though “grass” and “sky” have never been tagged to the same image by any users. We refer to this tagging approach as *collaborative image tagging* (CoIT). One of the challenging issues in CoIT is that it requires the existence of a reasonably large collection of user provided tags which however could be lacking initially in many cases (also known as the cold-start problem in CF [47]).

In this paper, we present a novel *Hybrid Probabilistic Model* (HPM) for automatic tag recommendation. HPM integrates low-level content descriptors of the image and the user provided high-level tags to take advantage of both content-based and collaborative approaches to image tagging. HPM formulates tagging as a tag prediction problem under a unified probabilistic framework. The visual content of an image is represented using a version of bag of visual words representation scheme and the user provided tags are leveraged by exploiting the tag correlations. HPM works as follows: if an image has user provided tags, HPM integrates the low-level image features and the user provided tags to predict additional tags for the image; if the image has never been tagged before, HPM essentially degenerates to content-based image labeling and will predict tags for the image based solely on the low-level image features. To utilize image contents for annotation, the estimation of the correlations between tags and visual features is always crucial. In this work, non-parametric kernel density estimation with an L_1 norm kernel function has been used. HPM explicitly exploits tag-to-tag correlation. As the number of images is typically very large and the tags provided by the users are highly diverse, the tag-image association matrix is often very sparse, making the estimation of the pairwise correlation of tags difficult and unreliable. To alleviate this inherent data sparsity problem in HPM, we developed a collaborative filtering method based on non-negative matrix factorization (NMF) for the correlation estimation. The contributions of this paper can be summarized as follows:

- A novel Hybrid Probabilistic Model (HPM) is developed to jointly model low-level image features and user provided tags for automatic image tagging.
- An NMF-based collaborative filtering method is developed to learn the tag to-tag correlation to address the data sparsity issue in HPM.

Experiments have been conducted on the popular Corel data sets, and a real-world image database [10] collected from Flickr. Our experimental results demonstrate that HPM is superior to conventional methods proposed for CoIT and CBIT in terms of tag recommendation accuracy. In the absence of user provided tags, HPM becomes a pure content-based annotation method. We show that

the performance of HPM is comparable to the best results reported in the content-based image annotation literature, which indicates that our new kernel-based learning method and the adopted image features are efficient and effective in estimating the correlations between visual features and semantic keywords. By using only a limited number (only 1 or 2) of user provided tags, the proposed HPM can dramatically boost the annotation performance of the pure content-based methods.

2 RELATED WORK

As previously discussed, this work is closely related to the area of CBIT. Also, the incorporation of user provided tags and their correlations makes previous works on collaboration filtering widely applicable. In this section, we briefly review current prevailing approaches to CBIT and CF.

2.1 Content-based Image Tagging

CBIT methods can be categorized into two main types, *generative models* and *classification models*. Generative models try to learn the joint probability distribution of semantic concepts and image visual features so that image annotation can be achieved using probabilistic inference. For instance, Duygulu et al. [16] considered image annotation as a machine translation problem and proposed a *translation model* (TM) between two discrete vocabularies: “blobs” which are generated by employing K -means algorithm to cluster segmented image regions and “words” appearing in the images’ captions. While TM considers a one-to-one relationship between image blobs and words, Jeon et al. [29] treated this as a cross-lingual retrieval problem and used a *cross-media relevance model* (CMRM) to estimate the joint probability distribution of words and image segments. In [32], Lavrenko et al. extended the CMRM and proposed the *continuous-space relevance model* (CRM) where image regions were represented by continuous-valued feature vectors rather than discrete blobs. Feng et al. [18] proposed a *multiple Bernoulli relevance model* (MBRM) which estimates the word probabilities using a multiple Bernoulli model and the image feature probabilities using a non-parametric kernel density estimation method. In [2], [3], Barnard et al. proposed the use of a *hierarchical aspect model* for modeling the semantic information provided by the associated text and image visual information. Blei and Jordan [6] proposed a Correlation Latent Dirichlet Allocation model (*correlation LDA model*) to relate words and images.

Unlike the generative model approach, the classification approach poses the task of CBIT as a supervised learning problem. The earliest efforts can be dated back to those focusing on detection of specific context of an image such as indoor/outdoor [50], landscape/cityscape [52], etc, where binary classifiers are typically trained using images within the context as positive examples and those out of the concept as

negative examples. The annotation task can further be formulated as a multi-class classification problem where each semantic keyword of interest defines an image class. Given an image, the probabilities that it belongs to different classes are computed and ranked. The top classes can then be suggested as the tags for the image. In particular, Li and Wang [36] proposed to construct a 2-D multi-resolution hidden Markov model (MHMM) for modeling each concept. Yavlinsky et al. [59] presented a simple Bayesian framework for CBIT using global image features. Carneiro et al. [8] formulated the annotation problem as a supervised multiclass labeling (SML) problem and computed the tag suggestion using multiple instance learning. Fan et al. [17] proposed a hierarchical classification framework to achieve automatic multi-level image annotation.

More recently, nearest neighbor models have been investigated in the annotation community with promising results. In particular, Makkadia et al. [43] have developed the joint equal distribution (JEC) technique where they used a combination of multiple features and distance metrics to find the nearest neighbors of the input image and transferred the tags of the nearest neighbors to the input image. Guillaumin et al. [25] have proposed the tag propagation (TagProp) method to annotate a input image by propagating the tags of the weighted nearest neighbors of that input image. The weighted nearest neighbors were identified by optimally integrating several image similarity metrics. Also, recent works have incorporated both contents and tag correlation for annotation. In particular, Loeff et al. used manifold regularization [44] and Xiang et al. adopted Markov Random Field (MRFA) [56] for this aim, respectively.

2.2 Collaborative Filtering

When one considers each image as a movie and the tag as a user rating, the automatic image annotation problem can readily be casted as collaborating filtering which is now commonly used for applications like movie and book recommendations. Collaborative filtering was first proposed for information filtering and recommendation of items (e.g., movies, music) whose detailed content descriptions are hard to acquire [27]. A number of memory-based and model-based methods have been proposed to exploit the correlation between user provided information, e.g., user ratings, usage patterns, etc. [7]. Recently, the collaborative filtering approach has also been applied to image retrieval for accumulating user feedbacks [51]. Such an approach can also be regarded as content-free because it does not directly look into the image content but rather purely operates on user feedbacks. While the CF approach does not require content-based analysis, high quality user provided information is needed at the very beginning before related systems can be deployed. This is the so-called cold start problem. To alleviate this, methods that combine content-based and CF approaches have been proposed (e.g., [4], [47]). This paper is in fact

making a contribution along this direction for automatic image annotation.

3 COLLABORATIVE IMAGE TAGGING

Tagging images using the collaborative approach relies on the assumption that images of a similar context should have more tags in common and can thus share their tags among themselves. We call this collaborative image tagging (CoIT) in the sequel. In this section, we formulate the CoIT problem, describe two possible solutions, and argue that both solutions can be integrated into a newly proposed hybrid probabilistic model to be presented in Section 4.

3.1 Problem Formulation

Let $\mathcal{I} = \{I_1, I_2, \dots, I_N\}$ be the set of images in the repository and $\mathcal{W} = \{w_1, w_2, \dots, w_M\}$ be the set of possible tags. Given that each image I_j is labeled with T_j tags $\{w_{u_1}, w_{u_2}, \dots, w_{u_{T_j}}\}$, the tagging records can be represented as an $M \times N$ tag-image association matrix V , where $v_{i,j} = 1$ indicates that the image I_j is labelled with the tag w_i and $v_{i,j} = 0$ means that the association is unknown. Among those associations with $v_{i,j} = 0$, there exist two possible cases. Either the image is not related to the tag or the tag should be there but is just missing due to various reasons. The CoIT problem is to predict the true values of those “missing” associations between the images and tags in a collaborative manner.

3.2 A Lightweight CF For Binary-Encoded Data

Memory-based collaborative filtering algorithms work because it has been observed that preference similarity between a pair of users can be revealed by the correlation between their ratings. Most of them assume each rating to be a numerical score and each item being rated to be of the same importance for characterizing the user. However, both assumptions do not fit well to the CoIT problem. On one hand, the tag-image association matrix V is binary-encoded. Also, it has been well known in the information retrieval community that words with different frequencies of occurrence carry different degrees of importance in the context being considered. The memory-based lightweight CF algorithm proposed in [55] takes care of the two issues just mentioned, making it suit particular well the characteristic of the CoIT problem. Let V be expressed as:

$$V = [v_1, \dots, v_N], \quad v_j = [v_{1,j}, \dots, v_{M,j}]^T, \quad j = 1, \dots, N$$

where T denotes matrix transpose. v_j can be interpreted as a binary encoded tag-based representation of the image I_j . The lightweight CF algorithm first defines a similarity measure between the binary representations of an image pair (I_p, I_q) , given as

$$\text{sim}(I_p, I_q) = \sum_{j=1}^M v_{j,p} v_{j,q} f_j, \quad (1)$$

Algorithm 1 Lightweight CF for CoIT

V – Tag-image association matrix of N training images and M tags

Input: v_p – Binary representation of test image I_p
 m – Neighborhood size
 K – Number of tags to be recommended

- 1: **for** $i = 1$ to M **do** $f_i = 1 + \frac{1}{n_i} \triangleright n_i$ is the frequency of the word w_i in the repository.
- 2: **end for**
- 3: **for** $q = 1$ to N **do**
- 4: $sim(I_p, I_q) = \sum_{j=1}^M v_{j,p} v_{j,q} f_j$
- 5: **end for**
- 6: $S_m(I_p) =$ Select m best “neighbors” with the highest similarity values $sim(I_p, I_q)$.
- 7: **for** $i = 1$ to M **do**
- 8: **if** $v_{i,p} == 1$ **then**
- 9: **continue;**
- 10: **else**
- 11: $\ell(w_k; I_p) = \sum_{I_j \in S_m(I_p)} v_{k,j} f_k$,
- 12: **end if**
- 13: **end for**
- 14: Return a tag recommendation list of K tags with the largest estimated association scores $\ell(w_k; I_p)$ to the test image I_p .

where f_j denotes the inverse frequency of the word w_j calculated using the equation

$$f_j = 1 + \frac{1}{n_j}, \quad (2)$$

and n_j denotes the number of the images which have been annotated with the word w_j in the repository. The idea of using the inverse word frequency to reduce the weighting of commonly occurring words is known to be effective, which captures the intuition that commonly occurring words are less useful in identifying the topic of an image than the rarely occurring ones. Also, as both $v_{j,p}$ and $v_{j,q}$ take only binary values, $sim(I_p, I_q)$ is equivalent to the number of tags which are common to both I_p and I_q and have the value discounted by f_j .

Based on (1), for each image I_i , one can compute its m best “neighbors”, denoted as $S_m(I_i)$. Then, the top- K tags to be recommended for I_i can be obtained by selecting the K tags which have the highest values of $\ell(w_k; I_i)$, given as

$$\ell(w_k; I_i) = \sum_{I_j \in S_m(I_i)} v_{k,j} f_k. \quad (3)$$

It has been demonstrated that treating less frequently occurring words as more important gives better performance than treating all words equally. The overall CF method for CoIT is summarized in Algorithm 1.

3.3 An NMF-based lightweight CF Algorithm

In contrast to other nearest neighbors algorithms which compute image similarity measured in some continuous

feature space, the lightweight CF algorithm, like other memory-based CF algorithms, consider images which share the same set of tags as neighbors. Its accuracy thus depends on the degree of sparsity of the tag-image association matrix V .

To alleviate this sparsity problem, one may resort to smoothing methods proposed for language modeling which discount the probabilities of the words seen in the matrix and assign the extra probability mass to the missing tags according to some “fallback” model [60]. Specifically, additive or Laplace smoothing simply adds an extra count to every word while Jelinek-Mercer linearly interpolates the maximum likelihood probability with the collection model to yield the smoothed estimation. Another promising way to alleviate the sparsity issue is to perform a low rank approximation. The fact that performing a low rank approximation can alleviate a similar sparsity problem in information retrieval can be dated back to the work of latent semantic indexing [14] where singular value decomposition (SVD) was used. Also, a probabilistic version of the latent semantic analysis (PLSA) was proposed in [28]. It has been shown that PLSA has higher modeling power and can outperform SVD. Non-negative matrix factorization (NMF), first proposed in [33] as another effective factorization method, was recently shown to be equivalent to PLSA [22] and particularly effective in addressing the sparsity problem in CF [61]. In this paper, we adopt NMF for smoothing V . It is noted that, independent from our work, Loeff and Farhadi [42] have also used matrix factorization for scene discovery where however SVD was used instead.

Given a tag-image association matrix $V_{M \times N}$ with $v_{i,j} \geq 0$ and a pre-specified positive integer $R < \min(M, N)$, NMF constructs two non-negative matrices $W_{M \times R}$ and $H_{R \times N}$ as the factors of V , such that $V \approx WH$, or $v_{i,j} \approx (WH)_{i,j} = \sum_{a=1}^R w_{i,a} h_{a,j}$. The solutions of W and H can be derived by solving the following optimization problem.

$$\min_{W, H} f(W, H) \equiv \frac{1}{2} \|V - WH\|_F^2, \quad s.t. \quad W \geq 0, H \geq 0. \quad (4)$$

The columns of W forms a set of R non-negative basis components. Each element of a basis component corresponds to a tag whose value indicates the “importance” of that tag in the component. Each column of H can be interpreted as the corresponding weighting coefficients. To perform NMF, multiplicative updating rules [34] are typically used to solve the optimization problem (4) in an iterative manner. To speed up the convergence, Lin [38] proposed the use of the projected gradient method which is adopted in this paper. In particular, (4) is replaced by two non-negative least squares, given as

$$\min_H \bar{f}(H) \equiv \frac{1}{2} \|V - WH\|_F^2, \quad s.t. \quad H \geq 0, \quad (5)$$

$$\min_W \bar{f}(W) \equiv \frac{1}{2} \|V^T - H^T W^T\|_F^2, \quad s.t. \quad W \geq 0. \quad (6)$$

which are then solved using the projected gradient method. Due to the local minimum problem, the solution obtained depends on the initialization of W and H . In our implementation, elements in W and H are initialized with values which are generated from a normal distribution with mean and standard deviation equal to 0 and 1 respectively. For more details about the projected gradient method, please refer to [38].

After the factorization step, the low rank approximation of \hat{V} computed by WH can thus be used as the “smoothed” tag-image association matrix and then fed to Algorithm 1 to yield the NMF-based lightweight CoIT algorithm which is summarized in Algorithm 2. In the next section, we show how this algorithm can be used to tackle the data sparsity issue in the estimation of tag-to-tag correlation.

Algorithm 2 NMF-based lightweight CoIT

V – Tag-image association matrix of N training images and M tags

v_p – Binary representation of test image I_p

Input: m – Neighborhood size

K – Number of tags to be recommended

R – Approximation matrices’ rank

ϵ – Tolerance for a relative stopping condition

- 1: Initialize W and H with non-negative values.
 - 2: Get the solutions of W and H by using the projected gradient method in [38] with the parameters of R , ϵ and maximum iterations.
 - 3: Take $\hat{V} = WH$ as the approximation of V .
 - 4: Put \hat{V} , v_p , m , K as the corresponding input of Algorithm 1.
 - 5: Return the output of Algorithm 1.
-

4 HYBRID PROBABILISTIC MODEL

Adopting only the CF approach to solve the CoIT problem ignores the fact that image visual contents should also play a crucial role in tag recommendation. Combining tags and image contents is a natural extension, which aims to not only solve the cold-start problem but also increase the overall accuracy. In this section, a unified probabilistic framework, named *hybrid probabilistic model* (HPM), is proposed for the integration.

4.1 The Probabilistic Framework

Given an image I and its existing set of tags \mathcal{U} , the posterior probability that a tag w should be assigned can be formulated as:

$$P(w|I, \mathcal{U}) = \frac{P(I|w)P(\mathcal{U}|w, I)P(w)}{P(I)P(\mathcal{U}|I)}. \quad (7)$$

For a given tag w , we can assume that \mathcal{U} and the test image I are independent. (7) can be re-written as

$$P(w|I, \mathcal{U}) = \frac{P(I|w)P(\mathcal{U}|w)P(w)}{P(I)P(\mathcal{U}|I)}. \quad (8)$$

Here, $P(w)$ refers to the prior probability of the tag w . $P(I|w)$ is the probability of generating the image I given the tag w . So, for instance, if w , the tag to be recommended, is “bridge”, we anticipate that the posterior probability of an image containing bridge-like visual features will be higher. And independently, we anticipate that the posterior probability of the user provided tags which often co-occurs with “bridge” (without considering the image context) should be higher. This assumption can make the problem much more tractable from the computational perspective and yet they are sufficiently accurate for the problem at hand, as supported by our empirical results to be presented in Section 6.

We further assume that for a given w the tags in \mathcal{U} are independent of each other, $P(\mathcal{U}|w)$ can be rewritten as

$$P(\mathcal{U}|w) = \prod_{t=1}^{|\mathcal{U}|} P(w_{i_t}|w), \quad (9)$$

where $P(w_{i_t}|w)$ is the probability of the occurrence of the tag w_{i_t} given the tag w . Putting everything together, and taking logarithm on (8), we have

$$\begin{aligned} \log P(w|I, \mathcal{U}) &= \log P(I|w) + \sum_{t=1}^{|\mathcal{U}|} \log P(w_{i_t}|w) \\ &\quad + \log P(w) - \log P(I) - \log P(\mathcal{U}|I). \end{aligned} \quad (10)$$

By computing the posterior probability for each tag $w \in \mathcal{V} \setminus \mathcal{U}$ according to (10), the tags can be ranked accordingly as suggestions to annotate the image. In our implementation, we assume that $P(I)$ and $P(\mathcal{U}|I)$ are constants across different $w \in \mathcal{V} \setminus \mathcal{U}$, and thus are ignored. $P(w)$ is estimated using the training set. The factors needed to be estimated are $P(I|w)$, the association between the visual contents and the tags, and $P(w_j|w_i)$, $i \neq j$, the tag correlation probability. Also, we need to select a particular model for representing images.

Before moving to the details for computing different factors in $P(w|I, \mathcal{U})$, it should be noted that our formulation does not consider the correlation of the w s which are recommended as new tags of an images, but assume them to be independent. That is, all the w s with the highest values of $P(w|I, \mathcal{U})$ will be recommended. One can also consider the correlation among the new tags. In [40], the correlation was factored in a progressive manner so as to make it computationally tractable. With a similar spirit, we could modify the tag recommendation as first picking the word $w^{(1)}$ with the largest probability $P(w^{(1)}|I, \mathcal{U})$ as the recommended tag and then picking the one $w^{(2)}$ with the largest value of $P(w^{(2)}|I, \{\mathcal{U}, w^{(1)}\})$ as the second tag, and so on. Our experimental results indicated that this progressive method achieved no significant performance gain and we therefore will not report results of this method.

4.2 Image as a Bag of Words

To represent the visual content of an image, we use the colored pattern appearance model (CPAM) which was

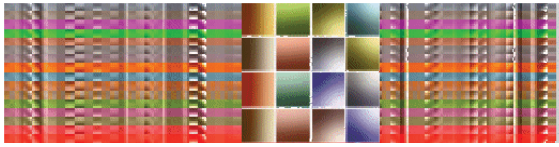


Fig. 1: Examples of CPAM appearance prototypes. Each prototype encodes certain chromaticity and spatial intensity patterns of a small image patch. Here the 16 patches in the middle are amplified prototypes randomly selected from 65,536 prototypes, a subset of them (4x4 pixels) are shown on both sides of 16 amplified patches.

proposed to capture both color and texture information of small patches in natural color images, and has been successfully applied to image coding, indexing and retrieval [46]. Although CPAM share some similarities with the “bag of visual words” idea [35] that is popular in the computer vision literature, CPAM does not only extract features around salient points, but also include features from the entire image and is therefore more suitable for our current application.

CPAM comes with a code book of common *appearance prototypes* which is built based on tens of thousands of image patches using vector quantization. Fig. 1 illustrates some examples of CPAM appearance prototypes². Given an image, a sliding window is used to decompose the image into a set of 4x4 tiles. Each small tile can then be encoded by one of the CPAM appearance prototypes that is most similar to the tile. The feature vector \mathbf{x} is built as a CPAM-based histogram for the image which tabulates the frequencies of the appearance prototypes being used to approximate (encode) every region of the pixels in the image. The CPAM-based feature vector \mathbf{x} comprises the achromatic spatial pattern histogram (ASPH) and chromatic spatial pattern histogram (CSPH). In our experiments (see Section 6), we select a code book with 64 achromatic prototypes and 64 chromatic prototypes and therefore each image is represented by a 128-dimensional feature vector. The distance between two CPAM-based feature vectors \mathbf{x}_m and \mathbf{x}_n is defined as

$$d(\mathbf{x}_m, \mathbf{x}_n) = \sum_{\forall i} \frac{|\text{ASPH}_m(i) - \text{ASPH}_n(i)|}{1 + \text{ASPH}_m(i) + \text{ASPH}_n(i)} + \sum_{\forall j} \frac{|\text{CSPH}_m(j) - \text{CSPH}_n(j)|}{1 + \text{CSPH}_m(j) + \text{CSPH}_n(j)}. \quad (11)$$

It is well known that L_1 norm is much more robust to outliers than L_2 norm. This is the distance used in the nonparametric kernel density estimation of $P(I|w)$ in the next subsection.

4.3 Estimating $P(I|w)$ using a Kernel-based Method

Let \mathbf{x} be the visual content representation of image I . The association between the visual content of I and the tag

w , i.e., $P(I|w)$, can be expressed as $P(\mathbf{x}|w)$. In order not to restrict the probability density distribution of $P(\mathbf{x}|w)$ to be of any parametric form [59], the non-parametric approach for density estimation is adopted. To achieve that by *kernel smoothing* [45], we define the multivariate kernel estimate of $P(\mathbf{x}|w)$ as

$$\hat{P}(\mathbf{x}|w) = \frac{1}{nC} \sum_{i=1}^n k\left(\frac{\mathbf{x} - \mathbf{x}_w^{(i)}}{h}\right), \quad (12)$$

where $\mathbf{x}_w^{(1)}, \dots, \mathbf{x}_w^{(n)}$ is the sample from the training set \mathcal{D}_w which includes feature vectors extracted from images tagged with the word w , $k(\cdot)$ is a kernel function that is placed over each point $\mathbf{x}_w^{(i)}$, $C = \int k(t)dt$ so that $\hat{P}(\mathbf{x}|w)$ integrates to one, and h is a positive scalar called bandwidth that reflects how wide a kernel is placed over each data point.

For the choice of the kernel function, a straightforward way is to define a d -dimensional Gaussian kernel [62], where d is the dimension of the feature vector \mathbf{x} . However, kernel smoothing may become less effective in high-dimensional spaces due to the problem known as the curse of dimensionality [21]. In this work, we define an L_1 kernel as

$$k_L(\mathbf{x}, \mathbf{x}^{(i)}; h) = \frac{1}{h} e^{-\frac{d(\mathbf{x}, \mathbf{x}^{(i)})}{h}}, \quad (13)$$

where $d(\mathbf{x}, \mathbf{x}^{(i)})$ is the distance measure between the feature vectors \mathbf{x} and $\mathbf{x}^{(i)}$ as defined in (11); and the bandwidth parameter h is chosen using the cross-validation method. The density function of \mathbf{x} given the tag w can now be estimated as

$$P(\mathbf{x}|w) \sim \hat{P}(\mathbf{x}|w) = \frac{1}{|\mathcal{D}_w|} \sum_{\mathbf{x}^{(i)} \in \mathcal{D}_w} k_L(\mathbf{x}, \mathbf{x}^{(i)}; h). \quad (14)$$

In our preliminary experiment, we found that this kernel gives satisfactory performance based on a number of data sets and thus is adopted in estimating the HPM model.

Modeling image features using kernel density estimation can also be found in other annotation related works, such as [18], [32], [59]. However, the practical challenges here are high computational and memory complexities. For large scale datasets, estimating the probability of an image with respect to all the training images can be very slow, hindering it from real-time application. Moreover, the entire training set has to be kept in memory at the testing time which consumes a large storage space. In the literature, different methods have been proposed for accelerating the kernel density estimation, e.g., Fast Gauss Transform [24], Improved Fast Gauss Transform [58], etc. In this paper, we propose to use k -d-tree [5] to speed up the estimation by identifying n nearest neighbors based on the empirical evidence presented in [15].

² Codebooks and Matlab code for CPAM are available at: <http://www.viplab.cs.nott.ac.uk/download/CPAM.html>

4.4 Tag Correlation Probability Estimation

To estimate the correlation between tags for enhancing image annotation accuracy, there exist a myriad of related efforts reported in the literature, including the use of the ontology WordNet [31] and the normalized Google distance (NGD) derived from the Word Wide Web [54]. However, they are application specific and their effectiveness is limited. For instance, with the use of WordNet, the semantic relations between words can be estimated from their glosses, hyponyms, hypernyms, meronyms, etc. However, the relations of concepts often go beyond those being defined in WordNet. For example, “doctor” and “hospital” do not share any hypernym or synonym in WordNet, but in reality they are highly related concepts. Regarding the use of NGD, as reported in [11], the unstructured nature of the Web makes direct use of NGD problematic in measuring word similarity.

Another approach for the estimation is to compute the correlation of words based on their occurrence, which is adopted in this paper. In the area of information retrieval, a related method called Automatic Local Analysis (ALA) [1], which makes use of word occurrence statistics in documents, is known to be effective for query expansion. It seems easy for the idea to be ported to image annotation as tag occurrence in images can be computed instead. However, it is unusual for a user to provide a large number of tags for each image. The tag-image association matrix, in many cases, is too sparse for any analysis to be effectively performed. This sparsity issue is also a well-known challenge in the area of CF.

4.4.1 NMF-Based Smoothing

Instead of using the sparse V matrix directly for ALA, we propose to represent each image in the training set with a *pseudo image profile* p_j which is defined as

$$p_{i,j} = \begin{cases} 1 & \text{if the } j\text{th image is labelled with the } i\text{th tag,} \\ \hat{v}_{i,j} & \text{otherwise.} \end{cases} \quad (15)$$

where $\hat{v}_{i,j}$ is the corresponding element of the approximation matrix $\hat{V} = WH$ as described in Section 3.3 (Elements whose values exceed 1 are set to 1.). Thus, one can interpret $\hat{v}_{i,j} \in [0, 1]$ as the confidence of the j th image to be tagged with the i th word. Compared with the matrix V , the tag-image association matrix P created based on the pseudo image profiles of all the training images is much denser.

4.4.2 Tag Correlation Estimation

Based on the matrix P , we can apply methods like ALA for the tag correlation estimation. Specifically, the correlation between tags w_i and w_j is defined as

$$\text{corr}(w_i, w_j) = \sum_{t=1}^N p_{i,t} \times p_{j,t}. \quad (16)$$

This measure essentially gives the estimated number of images where the two tags co-occur. As words that

appear very often in the data set tend to co-occur more frequently than most of the other words in the vocabulary, we thus normalize the measure by taking into account the word frequency, given as

$$P(w_j|w_i) = \frac{\text{corr}(w_i, w_j)}{\text{corr}(w_i, w_i) + \text{corr}(w_j, w_j) - \text{corr}(w_i, w_j)}, \quad (17)$$

for estimating the co-occurrence probability $P(w_j|w_i)$.

5 EXPERIMENTAL SETUP

In this section, we describe the experimental setup for assessing HPM’s performance, including the data sets used, the testing protocols, and the evaluation metrics.

5.1 Data Sets

Two standard Corel collections and a database of photos from Flickr were used in our experiments. The two Corel sets, named Corel5k and Corel30k, were both originated from the Corel stock photograph collection. They consist of all sorts of pictures, ranging from nature scenes to people portraits or sports photographs. Each picture is associated with a manually labeled textual caption that depicts the main objects appearing in the picture. The data set of photos from Flickr was collected from [10] (called NUS-WIDE in the sequel), which includes a set of images and the corresponding associated tags crawled from Flickr with the Flickr API.

The Corel5k data set contains 5,000 images and has been used in [8], [16], [18], [29], [32] for performance benchmarking. Each image is labeled with 1-5 words and there are altogether 371 distinct words in the vocabulary (the test set contains only 260 of the complete set). The average number of annotated words per image is 3.5, which reflects that this data set is indeed sparse. To make our results comparable to the mentioned benchmarks, we used the same training and test set partition as that in [16]. Out of the 5,000 images, 4,500 images form the training set and the remaining form the test set. To optimize the kernel bandwidth and rank R we further divide the training set into a training set of 4,000 images and a validation set of 500 images (see Table 1).

The Corel30k data set [8] is of similar nature as Corel5k except that it is substantially larger, containing 31,695 images and 5,587 words. We split the 31,695 images into training and testing at the ratio of 9:1. Only the words that are used by at least 10 images were selected to form the semantic vocabulary and there were 1,035 such words selected in total (950 of these appear in the test set). The average number of tags per image is 3.6. Corel30k due to its size, was included in our experiment to test the scalability of the proposed HPM. The statistics of the Corel30k data set are reported in Table 2.

The NUS-WIDE data set [10] contains 269,648 images collected from Flickr with a total of 425,059 unique tags. Among all the unique tags, there are 9,325 tags that occur more than 100 times. After removing some noisy tags such as those with wrong spelling, 5,018 unique

TABLE 1: Corel5k Statistics

	train	valid	test
# of images	4000	500	500
image size	384 × 256 or 256 × 384		
vocabulary size	371		
mean tag per image	3.52	3.52	

TABLE 2: Corel30k Statistics

	train	test
# of images	28,525 (90%)	3,170 (10%)
image size	365 × 237 or 237 × 365	
	384 × 256 or 256 × 384	
vocabulary size	1035	
mean tag per image	3.64	3.65

TABLE 3: NUS-WIDE Database Statistics

	train	test
# of images	161,789	107,859
image size	almost 240 × 160 or 160 × 240	
vocabulary size	5,018	

tags were finally left³. We randomly selected 161,789 images of this collection as the training set and the remaining as the test set (see Table 3). The tags in NUS-WIDE are collaboratively provided by a large group of heterogenous users. While most tags are correct, there are also many noisy and missing ones. This not so clean data set makes it good for evaluating the robustness of HPM.

5.2 Protocols

To demonstrate how effective HPM can exploit a small number of user provided tags to enhance the recommendation accuracy, we randomly selected $T(= 0, 1, 2)$ tags from each test image’s caption as the user provided tags and then attempted to predict the remaining ones. Following the conventions commonly used in the CF literature, we term these protocols *Given T* protocols [7]. For example, the *Given 1* protocol indicates that the user has already annotated the test images with one tag. The *Given 0* protocol means that there are no user provided tags available, and thus HPM becomes a pure content-based annotation system. We compared HPM with the conventional state-of-the-art content-based annotation models and a modified JEC model under these protocols.

5.3 Performance Metrics

We implemented HPM as well as some other conventional methods and assessed their annotation performance by computing their recall and precision rates per word in the test set. In particular, for a given word, let N_h be the number of images in the test set that are labelled with this tag by human, N_{sys} be the number of images that are suggested with this tag by our system, and N_c the number of images that our system gives correct

tag recommendation. The precision and recall rates are defined as $recall(w) = \frac{N_c}{N_h}$ and $precision(w) = \frac{N_c}{N_{sys}}$, respectively. For the two Corel data sets, under the protocol *Given 0*, similar to [8], [16], [18], [32], we annotated each test image with *five* words with the largest values computed according to (10). However, under the *Given 1* and *Given 2* protocols, the annotation lengths are set to be *four* and *three* respectively since the remaining tags of each test image is four and three at most. For the NUS-WIDE collection, N_{sys} can range from 1 to 15 (See Fig. 5). We then computed the average recall and precision rates over all the words in the test set. To ensure that our results under the *Given 1* and *Given 2* protocols are statistically reliable, we performed ten different runs for each of the experiments under these two protocols. And for each run, we randomly selected $T(= 1, 2)$ tags as the user provided tags. The results reported in the rest of the paper are the averages of ten trials.

In addition to the average word recall and precision rates, we also evaluated the coverage rate of words that the system has effectively learned, which provides an indication of the generalization ability of the system. The rate is calculated as the number of words with positive recall divided by the total number of words in the test set, which is denoted as “*Rate+*” for short. This metric is important because a biased model can also achieve relative high precision and recall rates by only performing extremely well on a small set of words [39].

6 EXPERIMENTAL RESULTS

In this section, we report experimental results. We first evaluate the effectiveness of the NMF-based smoothing technique and the L_1 -based kernel. Secondly, we compare the performance of HPM with CoIT. Thirdly, we compare HPM with a number of state-of-the-art content-based annotation methods and a modified JEC model. Finally, we give the time and space complexity of HPM.

6.1 Effectiveness of NMF-based Smoothing

We first present the experimental results on evaluating the effectiveness of the NMF-based method in alleviating the data sparsity problem which exists in both CoIT and HPM settings.

As mentioned in the previous section, the performance of collaborative approaches is greatly affected by how well they can handle the sparsity of the tag-image association matrix V . We conducted experiments on assessing CoIT with (Algorithm 2) and without (Algorithm 1) the use of NMF-based smoothing. We implemented the NMF-based lightweight CoIT (CoIT+NMF) with the rank R being set as 160 according to the validation set. Both of the lightweight CoIT and CoIT+NMF were implemented based on the best 30 neighbors. The evaluation results are tabulated in Table 4. According to the table, we see that the use of NMF-based smoothing can help boost the performance of the lightweight CoIT consistently with respect to the recall, precision and coverage rates.

3. <http://lms.comp.nus.edu.sg/research/NUS-WIDE.htm>

TABLE 4: Performance comparison of the CoIT algorithms with and without NMF for the Corel5k data set.

Protocols	Algorithms	Avg. Recall	Avg. Prec.	Rate ⁺
<i>Given 1</i>	CoIT	0.19	0.17	36.27%
	CoIT+NMF	0.22	0.19	40.18%
<i>Given 2</i>	CoIT	0.31	0.21	46.28%
	CoIT+NMF	0.35	0.24	50.23%

TABLE 5: Performance comparison of different smoothing techniques used for estimating tag co-occurrence probabilities in the HPM for the Corel5k data set.

Protocols	Algorithms	Avg. Recall	Avg. Prec.	Rate ⁺
<i>Given 1</i>	HPM+Laplace	0.38	0.24	58.92%
	HPM+JM	0.38	0.25	59.06%
	HPM+NMF	0.41	0.28	62.90%
<i>Given 2</i>	HPM+Laplace	0.45	0.28	62.52%
	HPM+JM	0.45	0.28	62.65%
	HPM+NMF	0.47	0.33	65.68%

For evaluating the proposed HPM which requires the estimation of tag co-occurrence probabilities (see (10)), we implemented three versions of HPM based on different smoothing techniques and performed experiments to compare their performance. Other than using NMF for estimating the tag co-occurrence probabilities (see Section 4.4.1), we also employed Laplace (HPM+Laplace) and Jelinek-Mercer (HPM+JM) [60] methods for the smoothing to avoid lots of zero entries. In Table 5, we tabulated the performances of HPM when different algorithms were used to estimate the word co-occurrences for the Corel5k data set. It is clearly seen that the NMF-based method is particularly effective in addressing the data sparsity problem and can lead to significant performance boosting. Compared with Laplace smoothing (HPM+Laplace), under *Given 1* protocol, the gains of HPM+NMF are 7.89% in recall, 16.67% in precision, and 6.75% in *Rate*⁺, respectively. Under *Given 2* protocol, similar performance boosting can be observed.

To further investigate how sensitive HPM is to the choice of low-rank R , we plot the overall annotation performance of HPM across different choices of R in Fig. 2. One can observe that HPM is not sensitive across a range of R values. Also, it is noted that the performance peaks when $R = 160$ which is also consistent to what being obtained using the validation set.

6.2 Effectiveness of L_1 -based kernel

As discussed in Sections 4.2 and 4.3, the effectiveness of kernel density estimation greatly relies on the choice of the kernel function. To assess the effectiveness of the proposed L_1 -based kernel, we compared it with the standard d -dimensional Gaussian kernel [62] and reported the results in Table 6. The L_1 -based kernel can significantly boost the annotation performance under all three protocols.

TABLE 6: Performance comparison between the L_1 based kernel and Gaussian kernel for the density estimation on Corel5k data set.

Protocols	Kernel	Avg. Recall	Avg. Prec.
<i>Given 0</i>	Gaussian	0.14	0.10
	L_1 -based	0.28	0.25
<i>Given 1</i>	Gaussian	0.28	0.18
	L_1 -based	0.41	0.28
<i>Given 2</i>	Gaussian	0.33	0.21
	L_1 -based	0.47	0.33

6.3 Comparison of HPM and CoIT

CoIT suggests tags based on only user provided tags and HPM does the same but based on both user provided tags and image visual features. As anticipated, and also depicted in Table 7, the average recall, precision and *Rate*⁺ results of HPM are found to be superior to those of the pure CF method, CoIT. Especially, under the *Given 0* protocol, CoIT is unable to suggest anything (the cold-start problem) while HPM still works well since it can make guesses based on the image visual content.

At this point, it is worth mentioning the capability of the NMF algorithm to identify the latent semantic groups of the data set, each of which, in this case, can be represented by images and their corresponding tags. Fig. 3 illustrates some of the extracted groups. In each group, the images with the largest elements in the same row of the matrix H generated by applying NMF to the tag-image association matrix of the Corel5k database are identified. The tags corresponding to the six largest elements of the particular component (i.e. the corresponding column) of the matrix W are also listed to explain the semantic meaning of that group. We can see that 1) pictures can fall into multiple groups, e.g. the second and forth (from left to right) pictures in Group 1 are also found in Group 2 and their topics can be interpreted as “arch” and “water”, respectively; and 2) some fine-grained topic modeling results are observed, e.g., Groups 3-5 are all about the topic of “bears” but further distinguished as “polar bear”, “black bear”, and “grizzly bear” respectively. Even though it is not surprising to find this byproduct since similar NMF-based analysis has been applied to application domains such as images [33] and documents [57]. Independent from our work, Loeff and Farhadi [42] have also used matrix factorization for scene discovery.

6.4 Comparison with State-of-the-art Results

In this section, we compare the performance of HPM with a few state-of-the-art content based annotation methods based on the Corel data sets. Also, we present results to demonstrate how HPM performs on the NUS-WIDE data set.

6.4.1 Performance on Corel5k Database

Performance comparison between HPM and a number of content-based annotation techniques evaluated in [8],

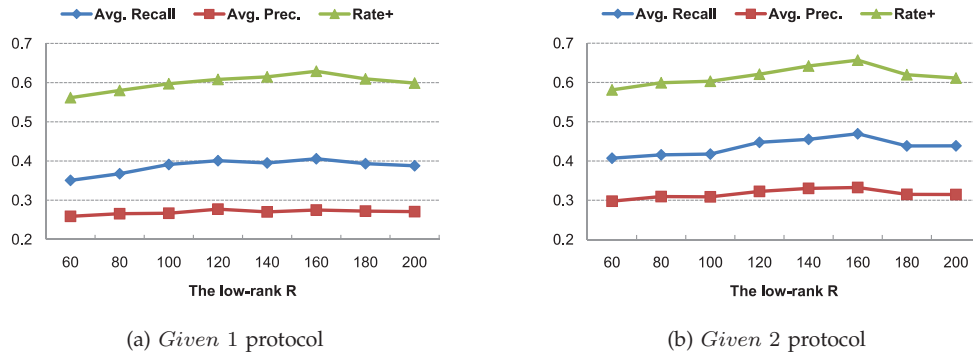


Fig. 2: The annotation performance of HPM across different low-rank R on Core5k database. (a) The annotation performance under *Given 1* protocol; (b) under *Given 2* protocol.

TABLE 7: Performance comparison between HPM and CoIT on the Core5k data set.

Models	CoIT			HPM		
	User tags			Visuality	User tags + Visuality	
	<i>Given 0</i>	<i>Given 1</i>	<i>Given 2</i>	<i>Given 0</i>	<i>Given 1</i>	<i>Given 2</i>
Avg. Recall	0	0.22	0.35	0.28	0.41	0.47
Avg. Prec.	0	0.19	0.24	0.25	0.28	0.33
$Rate^+$	0	40.18%	50.23%	52.31%	62.90%	65.68%

TABLE 8: Performance comparison between HPM and a few state-of-the-art content based annotation methods on the Core5k data set.

Cues	Models, Protocols	Recall	Prec.	$Rate^+$
Visuality	CRM [32]	0.19	0.16	41.15% (107/260)
	MBRM [18]	0.25	0.24	46.92% (122/260)
	SML [8]	0.29	0.23	52.69% (137/260)
	JEC [43]	0.32	0.27	53.46% (139/260)
	TagProp [25]	0.42	0.33	61.54% (160/260)
	HPM, <i>Given 0</i>	0.28	0.25	52.31% (136/260)
	JEC-CPAM, <i>Given 0</i>	0.26	0.21	48.46% (126/260)
Visuality + Tag correlation	Manifold [44]	0.40	0.21	N/A
	MRFA [56]	0.36	0.31	66.15% (172/260)
Visuality + User tags	JEC-CPAM, <i>Given 1</i>	0.31	0.22	48.05% (111.3/231.6)
	JEC-CPAM, <i>Given 2</i>	0.35	0.24	49.56% (93.7/189.1)
	HPM, <i>Given 1</i>	0.41	0.28	62.90% (143.8/228.6)
	HPM, <i>Given 2</i>	0.47	0.33	65.68% (121.2/184.6)

[18], [25], [32], [43] is tabulated in Table 8. They include CRM [32], MBRM [18], SML [8], JEC [43] and TagProp [25]. Also, we included two recent works which incorporate both contents and tag correlation using manifold regularization [44], Markov Random Field (MRFA) [56] respectively. Under the *Given 0* protocol (denoted as HPM, *Given 0*), as only low-level visual features are in use in the HPM, the performance of the proposed HPM is similar to SML, which proves that our kernel-based learning method and the adopted CPAM-based image features are efficient in estimating the associations between visual features and semantic concepts. Under *Given 0* protocol, HPM does not perform as well as TagProp which labels a query image by propagating the tags from its weighted nearest neighbors. When given only one user provided tag (HPM, *Given 1*), HPM achieves gains of 46.43 percent in recall, 12.00 percent in precision, and 20.24 percent in $Rate^+$ when compared

with the pure content-based method (HPM, *Given 0*). Under *Given 2* protocol, the proposed method (HPM, *Given 2*) achieves a consistent increase in performance which demonstrates that HPM effectively incorporates the user tags into the tag recommendation process. When compared with Manifold and MRFA, HPM-*Given 1* and HPM-*Given 2* also give comparable results.

To make a even fairer comparison with the state-of-the-art, we modified JEC to perform annotation under the *Given T* protocols but used the same CPAM-based features and L_1 distance metric as HPM. The new JEC method is denoted as JEC-CPAM. Under the *Given 0* protocol, we stuck to the same label transferring algorithm as in [43]. Under *Given 1* and *Given 2* protocols, suggested tags are ranked according to the combination of their co-occurrence with the given tags and local frequency (i.e. how often they appear in the nearest neighbors). In the implementation, the nearest neighbors are selected by taking the first 30 most similar images. From Table 8, it is seen that HPM outperformed JEC-CPAM under all three protocols, especially the *Given 1* and *Given 2* protocols. These results show that the proposed HPM framework can better exploit the two modalities of information (i.e. image content and given tags) to yield better annotation results.

Fig. 4 presents some examples of the annotations produced by HPM under the three protocols. Firstly, it is noted that HPM can often suggest plausible tags. For instance, the word “meadow”, though absent in the user provided tags, is identified by HPM and it is in fact a very suitable tag to describe Image 6. Secondly, the user provided tags are once again demonstrated to be able to enhance the annotation performance. Taking Image 3 as an example. Under *Given 0* protocol, “crowd” is a spurious annotation. However under *Given 1* protocol,

Group 1 Dominant tags	     	arch pair bridge water stone steel
Group 2 Dominant tags	     	water sky people bridge tree diver
Group 3 Dominant tags	     	SNOW polar ice bear face vehicle
Group 4 Dominant tags	     	black stream cheese hawk orchid bear
Group 5 Dominant tags	     	grizzly bear meadow marsh water grass

Fig. 3: Some semantic groups identified by applying NMF to the tag-image association matrix of the Corel5k data set. Each group is represented using not only images but also their associated tags for indicating the group topic. In particular, for each group, the presented images were identified by referring to the largest elements in the same row of the matrix H generated by NMF. The tags corresponding to the six largest elements of the particular component of the matrix W are listed. The font size indicates the relative dominance of the corresponding tag within a group.




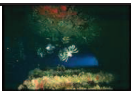






Image (1-5)					
Manual Annotation	sky jet plane smoke	bear polar snow tundra	tree forest cat tiger	coral fish lion cave	train tracks locomotive railroad
HPM, <i>Given 0</i> Annotation	smoke prop jet plane formation	tundra polar fog bear ice	crowd bengal stream tiger cat	cave fan fish coral reefs	buddhist locomotive railroad train door
HPM, <i>Given 1</i> Annotation	plane jet smoke f-16 prop	<u>bear</u> polar tundra snow ice	<u>cat</u> bengal tiger forest stream	<u>cave</u> fish coral reefs ocean	<u>train</u> locomotive railroad smoke tracks
HPM, <i>Given 2</i> Annotation	<u>plane</u> smoke jet f-16 prop	<u>bear</u> <u>tundra</u> polar snow ice	<u>tree</u> <u>cat</u> bengal tiger forest	<u>lion</u> <u>cave</u> fish coral reefs	<u>train</u> <u>locomotive</u> railroad smoke tracks
Image (6-10)					
Manual Annotation	field horses mare foals	flowers needles blooms cactus	tree plane herd zebra	water people pool swimmers	wall cars tracks formula
HPM, <i>Given 0</i> Annotation	mare foals horses field meadow	needles cactus angelfish blooms clearing	herd zebra plane stream white-tailed	butterfly pool swimmers athlete people	formula sails straightaway tracks cars
HPM, <i>Given 1</i> Annotation	<u>mare</u> foals horses field grass	<u>flowers</u> needles cactus blooms angelfish	<u>tree</u> zebra herd stream white-tailed	<u>water</u> pool swimmers people butterfly	<u>tracks</u> formula straightaway cars turn
HPM, <i>Given 2</i> Annotation	<u>field</u> <u>mare</u> foals horses grass	<u>needles</u> <u>cactus</u> blooms flowers saguaro	<u>tree</u> <u>plane</u> herd zebra stream	<u>water</u> <u>swimmers</u> pool butterfly people	<u>wall</u> <u>cars</u> formula straightaway tracks

Fig. 4: Illustration of some annotation results obtained using HPM under three protocols. The user provided tags are also shown. Regarding the *Given 1* and *Given 2* protocols, the underlined words are the user tags.

given “cat” as the user tag, “crowd” is replaced by the word “forest” since the word “forest” is more related to “cat” in this particular context as reflected by the co-occurrence correlation.

6.4.2 Performance on Corel30k Database

To evaluate HPM on larger scale annotation tasks, we compared its performance with that of SML (obtained

from [9]) on the Corel30k data set. Table 9 presents the comparison and shows that under *Given 0* protocol, the performance of HPM is comparable to that of SML with the average recall and precision rates slightly lower than those of SML. However, the number of words with non-zero recall obtained by HPM (439) is larger than that obtained by SML (424) which indicates a better generalization ability of HPM. Under *Given 1* and

TABLE 9: Performance comparison on Corel30k data set.

Cues	Visuality		User tags + Visuality	
	SML [8], [9]	HPM, <i>Given 0</i>	HPM, <i>Given 1</i>	HPM, <i>Given 2</i>
Avg. Recall	0.21	0.19	0.31	0.38
Avg. Prec.	0.12	0.10	0.16	0.19
<i>Rate</i> ⁺	44.63%	46.21%	55.71%	56.75%
	(424/950)	(439/950)	(488.6/877)	(423.8/747)

Given 2 protocols, the performance of HPM improves significantly and is consistent with that obtained based on the Corel5k data set. This shows that the HPM scales very well as the size of the data set increases.

6.4.3 Performance on NUS-WIDE Database

The NUS-WIDE database consists of 269,648 images. In our experiment, we used a test set with 1,000 tags. Note that CPAM is also used as the image features for this data set. Fig. 5 shows the results of HPM under different protocols on the NUS-WIDE database. Recall and precision are demonstrated with the annotation length N_{sys} ranging from 1 to 15. Under *Given 1* and *Given 2* protocols, i.e. test images have already been labelled with 1 and 2 user provided tags, the performances of HPM are superior to that of HPM under *Given 0* protocol which is a pure content-based system. For this very large real photo sharing database, the misspelled tags have been removed. Because the tags are provided by real world users in an uncontrolled manner, some of the tags are only weakly related to and some are even irrelevant to the visual contents of the photographs. These results demonstrate the robustness of our strategy in exploiting user provided high-level tags to boost the tag recommendation performance of the pure content-based system.

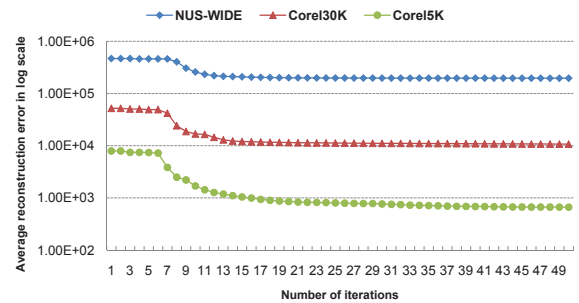
6.5 Computational Complexity of HPM

In this subsection, we analyze the computational complexity of HPM, including the off-line NMF-based smoothing and tag correlations estimation steps; and the online prediction process. All implementations were developed using Matlab and executed using an Intel Xeon 2.26 GHz PC with 4 GB memory.

Compared with simple smoothing methods such as Laplace and JM, NMF can achieve better annotation accuracy. The drawback of NMF is that it is computationally more complex. However, this step can be computed off-line which should make this less of a problem in real time implementation. Even though this step can be done off-line, it is still important to understand how long the off-line process will take. We conducted an experiment to study the issue. In particular, with the understanding that NMF implementation used for smoothing the tag-image association matrix can converge to different local minima, we applied NMF-based smoothing with 10 different random initializations and obtained the average runtime performance as shown in Table 10. The convergence of NMF indicated by the average reconstruction

TABLE 10: Average runtime performance of NMF on the three databases of 10 trials. The maximum number of iterations allowed is set to be 50.

Database	Size (M, R, N)	Run Time (in Sec.)
Corel5k	371 160 4,500	347
Corel30k	1,035 180 28,522	1,097
NUS-WIDE	1,000 200 156,585	2,360

Fig. 6: Average reconstruction errors ($\frac{1}{2}\|V - WH\|_F^2$) in log scale of NMF on the three data sets.

errors (computed by $\frac{1}{2}\|V - WH\|_F^2$) in log scale over iterations on all three data sets are plotted in Fig 6.

With the assumption that the NMF-based smoothing and tag correlation estimation steps can be done off-line for HPM, the performance of the real-time automatic annotation mainly rests on the efficiency of the tag prediction process of HPM. Referring to (10), the term with the dominating computational complexity is $P(I|w)$ which corresponds to the kernel estimation step. Assume that the size of the tag vocabulary is M and that each word can be tagged by at most the complete set of training images (with D training images), the complexity of HPM to annotate a new image will be $O(MD)$ times the complexity of computing a kernel distance. Fig. 7 presents the average time to tag an image and the total storage requirements for a Matlab instance to run HPM. To tag an image, we rank all the tags according to the posterior probability and then select the top- K tags to annotate that image. The average per-image tagging time is defined as the total time to annotate all the images in the test set divided by the number of the images in the test set. The results were empirically obtained by applying our implementation of HPM to the NUS-WIDE data set. The annotation time includes the time spent on the feature extraction and the tag prediction, but the former one is essentially negligible when compared with the latter. We demonstrated that the annotation time and memory storage increase linearly as the value of MD

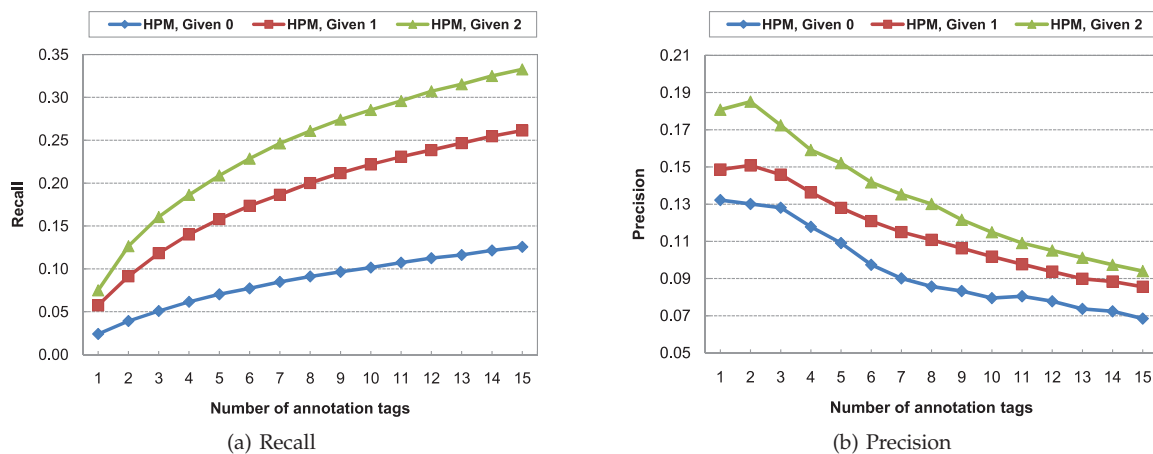


Fig. 5: Performance of HPM under different protocols based on the NUS-WIDE database with a test set of 1,000 tags.

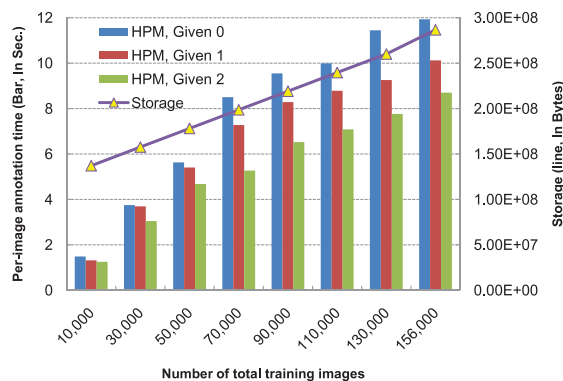


Fig. 7: Average per-image tagging time (bar, units indicated in the left y-axis) and storage requirements (line, units indicated in the right y-axis) of HPM under different protocols on the NUS-WIDE data set.

increases.

7 CONCLUSIONS

In this paper, we have presented a novel hybrid probabilistic model (HPM) to jointly model the correlations between low-level image features and high level semantic tags and the correlations between high level semantic tags to automatically tag images. As well as presenting the HPM framework, we have developed NMF-based collaborative filtering method to address the data sparsity issues in the estimation of the tag to tag correlations and an effective L_1 norm kernel method to estimate the probability density functions of low-level image features. We have presented extensive experimental results which show that for tagging images without any existing labels, the new HPM framework can do as well as the best techniques in the literature. We have also shown that by using one or two user provided tags, the HPM can effectively recommend more tags for the images by fully exploiting the correlations between low-level features

and high level tags and the correlations between the high level tags. The HPM can be easily incorporated into an image tagging tool where for a given image, the user can first manually tag the image with one or two labels, and then the HPM will take over and based on the user provided tags and the images low-level contents to automatically recommend more tags for the image. Building such a tool will make image labeling more efficient, less labor intensive, and ultimately help making the huge number of images on the Internet more easily searchable.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers for their insightful comments which helped to improve the paper. They also gratefully thank Dr. Xin Li of Hong Kong Baptist University for helpful and informative discussions on non-negative matrix factorization and its efficient implementation for large scale datasets. This work was supported in part by 973 Program (No.2010CB327900), NSF of China (No. 60873178) and Shanghai Leading Academic Discipline Project (No. B114).

REFERENCES

- [1] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison Wesley, May 1999.
- [2] Kobus Barnard, Pinar Duygulu, David A. Forsyth, Nando de Freitas, David M. Blei, and Michael I. Jordan. Matching words and pictures. *Journal of Machine Learning Research*, 3:1107–1135, 2003.
- [3] Kobus Barnard and David A. Forsyth. Learning the semantics of words and pictures. In *Proc. of IEEE Intl. Conf. on Computer Vision (ICCV'01)*, pages 408–415, 2001.
- [4] Justin Basilico and Thomas Hofmann. Unifying collaborative and content-based filtering. In *Proc. of 21st Intl. Conf. on Machine Learning (ICML'04)*, 2004.
- [5] Jon Louis Bentley. Multidimensional binary search trees used for associative searching. *Commun. ACM*, 18:509–517, September 1975.
- [6] David M. Blei and Michael I. Jordan. Modeling annotated data. In *Proc. of the 26th Annual Intl. ACM SIGIR Conference (SIGIR'03)*, pages 127–134, 2003.

- [7] John S. Breese, David Heckerman, and Carl Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI'98)*, pages 43–52, 1998.
- [8] G. Carneiro, A. B. Chan, P. J. Moreno, and N. Vasconcelos. Supervised learning of semantic classes for image annotation and retrieval. *IEEE Trans Pattern Anal Mach Intell*, 29(3):394–410, March 2007.
- [9] A. B. Chan, P. J. Moreno, and N. Vasconcelos. Using statistics to search and annotate pictures: an evaluation of semantic image annotation and retrieval on large databases. In *Proceedings of the American Statistical Association*, Seattle, August, 2006.
- [10] Tat-Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, and Yan-Tao. Zheng. Nus-wide: A real-world web image database from national university of singapore. In *Proc. of ACM Conf. on Image and Video Retrieval (CIVR'09)*, Santorini, Greece., July 8–10, 2009.
- [11] Rudi Cilibrasi and Paul M. B. Vitányi. The google similarity distance. *IEEE Trans. Knowl. Data Eng.*, 19(3):370–383, 2007.
- [12] Ritendra Datta, Dhiraj Joshi, Jia Li, and James Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Comput. Surv.*, 40(2), 2008.
- [13] Ritendra Datta, Dhiraj Joshi, Jia Li, and James Ze Wang. Tagging over time: real-world image annotation by lightweight meta-learning. In *Proc. of ACM Multimedia*, pages 393–402, 2007.
- [14] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41:391–407, 1990.
- [15] Mike Klaas Dustin Lang and Nando de Freitas. Empirical testing of fast kernel density estimation algorithms. Technical Report UBC TR-2005-03, The University of British Columbia Computer Sciences Department, March 2005.
- [16] P. Duygulu, K. Barnard, J. de Freitas, and D. Forsyth. Object recognition as machine translation :learning a lexicon for a fixed image vocabulary. In *Proc. of the 7th European Conference on Computer Vision (ECCV'02)*, 2002.
- [17] Jianping Fan, Yuli Gao, and Hangzai Luo. Hierarchical classification for automatic image annotation. In *SIGIR*, pages 111–118, 2007.
- [18] S. Feng, V. Lavrenko, and R. Manmatha. Multiple bernoulli relevance models for image and video annotation. In *Proc. IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR'04)*, volume 2, pages 1002–1009, 2004.
- [19] Shaolei Feng and R. Manmatha. A discrete direct retrieval model for image and video retrieval. In *Proceedings of the 2008 international conference on Content-based image and video retrieval, CIVR '08*, pages 427–436, New York, NY, USA, 2008. ACM.
- [20] Flickr. <http://www.flickr.com>, Yahoo!, 2005.
- [21] J. Friedman, W. Stuetzle, and A. Schroeder. Projection pursuit density estimation. *Journal of the American Statistical Association*, (79):599–608, 1984.
- [22] Eric Gaussier and Cyril Goutte. Relation between pls and nmf and implications. In *In Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval, (SIGIR'05)*, pages 601–602, New York, NY, USA, 2005. ACM.
- [23] David Goldberg, David Nichols, Brian M. Oki, and Douglas B. Terry. Using collaborative filtering to weave an information tapestry. *Commun. ACM*, 35(12):61–70, 1992.
- [24] Leslie Greengard and John Strain. The fast gauss transform. *SIAM J. Sci. Comput.*, 2:79–94, 1991.
- [25] Matthieu Guillaumin, Thomas Mensink, Jakob Verbeek, and Cordelia Schmid. Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation. In *International Conference on Computer Vision*, pages 309–316, sep 2009.
- [26] C. Halaschek-Wiener, J. Golbeck, A. Schain, M. Grove, B. Parsia, and J. Hendler. Photostuff—an image annotation tool for the semantic web. In *Proc. of 4th Intl. Semantic Web Conference*, 2005.
- [27] Jonathan L. Herlocker, Joseph A. Konstan, Al Borchers, and John Riedl. An algorithmic framework for performing collaborative filtering. In *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 230–237, New York, NY, USA, 1999. ACM Press.
- [28] Thomas Hofmann. Probabilistic latent semantic analysis. In *In Proc. of Uncertainty in Artificial Intelligence, (UAI'99)*, pages 289–296, 1999.
- [29] Jiwoon Jeon, Victor Lavrenko, and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. In *Proc. of the 26th Annual Intl. ACM SIGIR Conference (SIGIR'03)*, pages 119–126, 2003.
- [30] Rong Jin, Joyce Y. Chai, and Luo Si. Effective automatic image annotation via a coherent language model and active learning. In *Proc. of ACM Multimedia*, pages 892–899, 2004.
- [31] Yohan Jin, Latifur Khan, Lei Wang, and Mamoun Awad. Image annotations by combining multiple evidence & wordnet. In *Proc. of the 13th annual ACM international conference on Multimedia*, Hilton, Singapore Nov. 06–11, 2005.
- [32] V. Lavrenko, R. Manmatha, and J. Jeon. A model for learning the semantics of pictures. In *Proc. of Advances in Neural Information Processing Systems (NIPS'03)*, 2003.
- [33] Daniel D. Lee and H. Sebastian Seung. Learning the parts of objects by nonnegative matrix factorization. *Nature*, (401):788–791, 1999.
- [34] Daniel D. Lee and H. Sebastian Seung. Algorithms for non-negative matrix factorization. In *NIPS*, pages 556–562, 2000.
- [35] Fei-Fei Li, Pietro Perona, and California Institute of Technology. A bayesian hierarchical model for learning natural scene categories. In *Proc. of IEEE Intl. Conf. on Computer Vision and Pattern Recognition (CVPR'2005)*, pages 524–531, 2005.
- [36] Jia Li and James Ze Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(9):1075–1088, 2003.
- [37] Jia Li and James Ze Wang. Real-time computerized annotation of pictures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(6):985–1002, 2008.
- [38] Chih-Jen Lin. Projected gradient methods for nonnegative matrix factorization. *Neural Computation*, 19(10):2756–2779, 2007.
- [39] Jing Liu, Bin Wang, Mingjing Li, Zhiwei Li, Wei-Ying Ma, Hanqing Lu, and Songde Ma. Dual cross-media relevance model for image annotation. In *Proc. of ACM Multimedia*, pages 605–614, 2007.
- [40] Jing Liu, Bin Wang, Hanqing Lu, and Songde Ma. A graph-based image annotation framework. *Pattern Recognition Letters*, 29(4):407–415, 2008.
- [41] Wenyin Liu., S. Dumais, Y. Sun, H. Zhang, M. Czerwinski, and B. Field. Semi-automatic image annotation. In *INTERACT2001, 8th IFIP TC.13 Conference on Human-Computer Interaction*, Tokyo, Japan Jul. 9–13, 2001.
- [42] Nicolas Loeff and Ali Farhadi. Scene discovery by matrix factorization. In *ECCV (4)*, pages 451–464, 2008.
- [43] Ameesh Makadia, Vladimir Pavlovic, and Sanjiv Kumar. A new baseline for image annotation. In *ECCV (3)*, pages 316–329, 2008.
- [44] Ian Endres Nicolas Loeff, Ali Farhadi and David A. Forsyth. Unlabeled data improves word prediction. In *ICCV*, 2009.
- [45] E Parzen. On estimation of a probability density and mode. *Annals of Mathematical Statistics*, 35:1065–1076.
- [46] G. Qiu. Indexing chromatic and achromatic patterns for content-based colour image retrieval. *Pattern Recognition*, 35:1675–1685, August 2002.
- [47] A. Schein, A. Popescul, L. Ungar, and D. Pennock. Methods and metrics for cold-start recommendations. In *Proc. of the 25th Annual Intl. ACM SIGIR Conf. (SIGIR'02)*, pages 253–260, 2002.
- [48] Börkur Sigurbjörnsson and Roelof van Zwol. Flickr tag recommendation based on collective knowledge. In *Proceedings of the 17th International Conference on World Wide Web (WWW'2008)*, Beijing, China, April 21–25, pages 327–336, 2008.
- [49] A. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(12):1349–1380, 2000.
- [50] Martin Szummer and Rosalind W. Picard. Indoor-outdoor image classification. In *CAIVD*, pages 42–51, 1998.
- [51] Shingo Uchihashi and Takeo Kanade. Content-free image retrieval by combinations of keywords and user feedbacks. In *Proc. of the 4th Intl. Conf. on Image and Video Retrieval (CIVR'05)*, pages 650–659, 2005.
- [52] Aditya Vailaya, Anil K. Jain, and HongJiang Zhang. On image classification: city images vs. landscapes. *Pattern Recognition*, 31(12):1921–1935, 1998.
- [53] Nuno Vasconcelos and Murat Kunt. Content-based retrieval from image databases: current solutions and future directions. In *Proc. of IEEE Intl. Conf. on Image Processing*, pages 6–9, 2001.

- [54] Yong Wang and Shaogang Gong. Refining image annotation using contextual relations between words. In *Proc. of the 6th Intl. Conf. on Image and Video Retrieval (CIVR'07)*, pages 425–432, 2007.
- [55] Sholom M. Weiss and Nitin Indurkha. Lightweight collaborative filtering method for binary-encoded data. In *Proceedings of the 5th European Conference on Principles of Data Mining and Knowledge Discovery (PKDD 2001)*, pages 484–491, Sep. 03–05, 2001.
- [56] Yu Xiang, Xiangdong Zhou, Tat-Seng Chua, and Chong-Wah Ngo. A revisit of generative model for automatic image annotation using markov random fields. In *CVPR*, pages 1153–1160, 2009.
- [57] Wei Xu, Xin Liu, and Yihong Gong. Document clustering based on non-negative matrix factorization. In *SIGIR*, pages 267–273, 2003.
- [58] Changjiang Yang, Ramani Duraiswami, Nail A. Gumerov, and Larry S. Davis. Improved fast gauss transform and efficient kernel density estimation. In *ICCV*, pages 464–471, 2003.
- [59] Alexei Yavlinsky, Edward Schofield, and Stefan M. Ruger. Automated image annotation using global features and robust non-parametric density estimation. In *Proc. of the 4th Intl. Conf. on Image and Video Retrieval (CIVR'05)*, pages 507–517, 2005.
- [60] ChengXiang Zhai and John D. Lafferty. A study of smoothing methods for language models applied to information retrieval. *ACM Trans. Inf. Syst.*, 22(2):179–214, 2004.
- [61] Sheng Zhang, Weihong Wang, James Ford, and Fillia Makedon. Learning from incomplete ratings using non-negative matrix factorization. In *Proceedings of the Sixth SIAM International Conference on Data Mining, April 20-22, 2006, Bethesda, MD, USA, 2006*.
- [62] Ning Zhou, William K. Cheung, Xiangyang Xue, and Guoping Qiu. Collaborative and content-based image labeling. In *ICPR*, pages 1–4, 2008.



about his research can be found in: <http://www.viplab.cs.nott.ac.uk/>.



Guoping Qiu received the B.Sc. degree in Electronic Measurement and Instrumentation from the University of Electronic Science and Technology of China in 1984 and the Ph.D. degree in Electrical and Electronic Engineering from the University of Central Lancashire, Preston, UK, in 1993. He is currently a Reader in the School of Computer Science, University of Nottingham, UK. He has research interests in the broad area of computational visual information processing and has published widely in this area. More

Xiangyang Xue received the B.S., M.S., and Ph.D. degrees in communication engineering from Xidian University, Xi'an, China in 1989, 1992 and 1995, respectively. He joined the Department of Computer Science, Fudan University, Shanghai, in May 1995, where he is currently a professor. His research interests include multimedia information processing, retrieval and filtering, pattern recognition and machine learning. He has published more than 100 research papers in journals or conference proceedings.



Ning Zhou received the BSc degree from Sun Yat-sen University, Guangzhou, China in 2006 and MSc degree from Fudan University, Shanghai, China in 2009, both in computer science. He is currently a PhD student in the Department of Computer Science, the University of North Carolina at Charlotte. His current research interests include computer vision, machine learning with applications to image annotation, image retrieval and collaborative information filtering.



William K. Cheung received the BSc and MPhil degrees in electronic engineering from the Chinese University of Hong Kong and the PhD degree in computer science in 1999 from the Hong Kong University of Science and Technology. He is an associate professor in the Department of Computer Science, Hong Kong Baptist University. He has served as the co-chair and a program committee member of a number of international conferences, as well as a guest editor of journals on areas including artificial intelligence,

Web intelligence, data mining, Web services, and e-commerce technologies. Also, he has been on the editorial board of the IEEE Intelligent Informatics Bulletin since 2002. His recent research interests include collaborative information filtering, web and text mining, distributed and privacy-preserving data mining, planning under uncertainty, and process mining.