Image Retrieval under Very Noisy Annotations

Kazuya Ueki, Tetsunori Kobayashi
Faculty of Science and Engineering, Waseda University,
Room 40-701, Waseda-machi 27, Shinjuku-ku, Tokyo, 162–0042 Japan
ueki@pcl.cs.waseda.ac.jp

Abstract—**In recent years, a significant number of tagged images uploaded onto image sharing sites has enabled us to create high-performance image recognition models. However, there are many inaccurate image tags on the Internet, and it is very laborious to investigate the percentage of tags that are incorrect. In this paper, we propose a new method for creating an image recognition model that can be used even when the image data set includes many incorrect tags. Our method has two superior features. First, our method automatically measures the reliability of annotations and does not require any parameter adjustment for the percentage of error tags. This is a very important feature because we usually do not know how many errors are included in the database, especially in actual Internet environments. Second, our method iterates the error modification process. It begins with the modification of simple and obvious errors, gradually deals with much more difficult errors, and finally creates the high-performance recognition model with refined annotations. Using an object recognition image database with many annotation errors, our experiments showed that the proposed method successfully improved the image retrieval performance in approximately 90 percent of the image object categories.**

1. Introduction

In the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2012, the winning team used convolutional neural networks (CNNs) and overwhelmingly improved on the best performance in the object recognition task [1], and there has still been tremendous improvement each year since then. One of the main contributions of this breakthrough has been the development of a large-scale image database called ImageNet. In order to prepare the database, images on image sharing sites are manually annotated using a crowdsourcing service such as Amazon Mechanical Turk. The ImageNet database used in the ILSVRC was also created by the crowdsourcing service. Currently, a large number of tagged images are uploaded to social media sites such as Flickr, Instagram, and Facebook. The number of photos uploaded onto social networks is rapidly increasing, and now nearly 2 billion photos are shared daily across the social networks. Some image data are tagged by users, and the performance of recognition models is thus expected to be further improved by using these data. However, some of these tags are not correct [1] and have been deliberately created by ill-disposed users, so it is very daunting to annotate the increasing number of images day by day. Eradicating the incorrect annotations is not an easy task even though the annotation is done by the crowdsourcing service. If the inaccurate annotations can be corrected, the performance of image recognition is promising. Therefore, correction of annotation errors is an important task for improving image recognition performance. For these reasons, we propose a method that can create a noise-robust image recognition model through annotation refinement even under conditions of very noisy annotations.

This paper is organized as follows: In Section 2, we describe related research. In Section 3, we explain the problem setting. In Section 4, we present the proposed method. In Section 5, we discuss experiments that use an object recognition database with inaccurate annotations. In Section 6, we give our conclusions and suggestions for future research.

2. Related Work

Today, many digital images and videos are uploaded to social networks, and tagging and commenting on this content are becoming more common. This leads to more effective image retrieval. Thus, there have been many research papers in recent years on the use of image tags [3]. Most research deals with the three closely related problems of image tag assignment, image tag refinement, and tag-based image retrieval. The problem of image tag assignment is defined as follows: given an unlabeled image, assign a number of tags related to the image content [4], [5], [6], [7]. The problem of image tag refinement is defined as follows: given an image associated with some initial tags as annotated by users, remove incorrect tags and sometimes modify them [8], [9], [10], [11], [12]. The problem of tag-based image retrieval is defined as follows: given a collection of images annotated with tags, retrieve images that are relevant with respect to the input keyword [9], [13], [14], [15], [16]. In this paper, we deal with the problems of image tag refinement and tag-based image retrieval.

Additionally, different types of modalities are used in existing studies: 1) tag based [17], [18]; 2) tag and image

[1] In Flickr, only about 50% of tags are actually relevant to the images [2].

978-0-9928-6265-7/16/$31.00 ©2016 IEEE

based [2], [4], [5], [19], [20], [21]; and 3) tag, image, and user information based [13], [22]. This paper as well uses the most common combination: 2) tag and image based. In particular, our method is intimately related to model-based approaches [19], [20], [21]. However, none of the existing methods can cope with very noisy annotations. They need to manually tune many parameters depending on the proportion of annotation errors: if the database is changed, optimization of parameters is needed to create a high-performance image recognition model.

For these reasons, we propose a new method that can create a high-performance recognition model through an iterative annotation refinement process by estimating the reliability of annotations even when there are many irrelevant annotations.

3. Establishment of Problem

3.1. Creation of a Recognition Model under Many Annotation Errors

We consider a problem of keyword-based image retrieval from a massive image database, namely, for a user inputting a keyword to acquire appropriate candidate images relevant to the input keyword. In order to achieve this, we create a recognition model for each respective category (keyword), one that can output scores or likelihoods. Training image samples are annotated as positive or negative for each category. However, in this paper, we consider the situation where many tags are not correctly annotated; for example, some positive samples are annotated as negative, and some negative samples are annotated as positive.

3.2. Evaluation Criteria

The evaluation criterion for the image recognition model is discussed in this subsection. First, a prediction score is calculated by inputting every test image sample into the created recognition model of each category. Then, image samples are sorted by their scores in descending order, and a ranked list is created. Here, the average precision of each category is defined as:

$$AP = \frac{1}{N^{(\mathrm{te})}_{\mathrm{pos}}} \sum_{i=1}^{N^{(\mathrm{te})}} P_i \cdot Rel_i, \tag{1}$$

where $N_{(\mathrm{te})}$ is the number of test images, $N^{(\mathrm{te})}_{\mathrm{pos}}$ is the number of positive test images, i is the rank in the ordered list of results retrieved from $N_{(\mathrm{te})}$ images, P_i is defined as precision computed at the i-th rank, and Rel_i takes on the value 1 or 0, representing being relevant or irrelevant, respectively. Average precision measures how well a recognition model retrieves relevant images at the top of the ranked retrieval results.

Finally, the evaluation of the recognition model is performed using the mean average precision (mAP): the average-precision scores averaged across all categories in the image database.

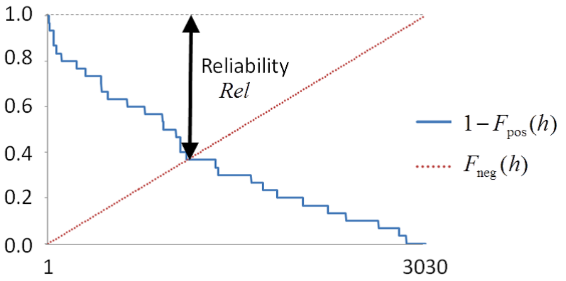

Figure 1. The reliability of annotations calculated from the cumulative distribution functions of the positive (pos) and negative (neg) annotations. The $F_{\mathrm{neg}}(h)$ curve looks linear. This is because the number of negative annotations (3,000) is vastly larger than the number of positive annotations (30).

4. Proposed Method

4.1. Characteristics of Proposed Method

Our proposed method has two distinct features.

The first is that it has a function to iteratively correct irrelevant annotations. If there are many images incorrectly annotated, it is very hard to modify all the errors at once. By starting with an easier problem and incrementally increasing the difficulty, more complicated annotation errors are expected to be corrected.

The second is that by automatically calculating the reliability of annotations, our method does not require trial-and-error manual parameter adjustment. This feature is very convenient in actual Internet environments where we do not know the percentage of annotation errors that are included in the database. Our proposed method can judge when the iterative process should stop by utilizing an automatic estimation of the reliability of annotations.

4.2. Definition of the Reliability of a Recognition Model

Here, we define the **reliability** of annotations as used to achieve the two superior characteristics discussed above. The reliability is determined using the distributions of positive and negative annotations on the ranked list sorted by scores estimated through the cross-validation. Specifically, it is calculated using the following procedure:

1) Calculate predicted scores for all the training data $X = \{(\boldsymbol{x}_1, l_1), \ldots, (\boldsymbol{x}_n, l_n)\}$ based on K-fold cross-validation, where $\boldsymbol{x}_i$ is the image feature vector, and l_i is the annotation denoted as $l_i = positive$ for the positive annotation and $l_i = negative$ for the negative annotation. Repeat this score calculation P times, and obtain all the scores $s_i (i = 1, \cdots, n)$ by taking the average of the P trials.
2) Sort the training image samples by score in descending order.

```
Algorithm: Method for creating a recognition model for each category
Input:    A set of training images: $X = \{(\boldsymbol{x}_1, l_1), \ldots, (\boldsymbol{x}_n, l_n)\}$
Output:   A recognition model: $M$
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
Step 1:   Initialize the maximum reliability of annotations: $Rel_{\max} = 0$.
Step 2:   Initialize the iteration counter: $r = 1$.
Step 3:   loop
              =====↓↓↓ Procedure for the elimination of unreliable annotations ↓↓↓=====
Step 4:       Initialize scores $s_i = 0.0$ $(i = 1, \ldots, n)$.
Step 5:       for $p = 1 \rightarrow P$
Step 6:           Calculate scores of all the training images $X$ by $K$-fold cross-validation.
Step 7:       Calculate the average scores $s_i$ of the $P$ trials.
Step 8:       Sort all the training data $X$ by score in descending order.
Step 9:       Calculate the reliability $Rel_r$ by Equation (4).
Step 10:      if $Rel_r > Rel_{\max}$ then
Step 11:          $Rel_{\max} = Rel_r$
Step 12:      else
Step 13:          return  $M_{r-1}$
Step 14:      Eliminate unreliable annotations using the criterion defined in Subsection 4.3: $l_i = unknown$.
              =====↓↓↓ Procedure for creating and evaluating a recognition model ↓↓↓=====
Step 15:      Train a recognition model $M_r$ using training samples excluding data with $l_i = unknown$.
Step 16:      If needed, evaluate the recognition model $M_r$.
              =====↓↓↓ Procedure for re-annotation ↓↓↓=====
Step 17:      Initialize scores for data annotated as $l_i = unknown$: $s_i = 0.0$.
Step 18:      Calculate scores $s_i$ by inputting data with $l_i = unknown$ into the recognition model $M_r$.
Step 19:      For the image data annotated as $l_i = unknown$,
              assign $l_i = positive$ for $s_i \geq 0$ and $l_i = negative$ for $s_i < 0$.
              =====↓↓↓ Increment the iteration count ↓↓↓=====
Step 20:      $r = r + 1$
Step 21:  end loop
```

Figure 2. The method for creating a refined recognition model under very noisy annotations.

3) From the highest score to the lowest score, calculate the value of the cumulative distribution function of positive annotations at the h-th rank by

$$F_{\text{pos}}(h) = \frac{\#pos(h)}{\#pos(X)}, \tag{2}$$

where $\#pos(X)$ is the total number of training data annotated as positive, and $\#pos(h)$ is the number of data annotated as positive from the top to the h-th rank.

4) In a similar manner, from the highest score to the lowest score, calculate the value of the cumulative distribution function of negative annotations at the h-th rank by

$$F_{\text{neg}}(h) = \frac{\#neg(h)}{\#neg(X)}, \tag{3}$$

where $\#neg(X)$ is the total number of training data annotated as negative, and $\#neg(h)$ is the number of data annotated as negative from the top to the h-th rank.

5) Calculate the **reliability** using the average value of $1 - F_{\text{pos}}(h')$ and $F_{\text{neg}}(h')$ at the h'-th rank, where the two functions $1 - F_{\text{pos}}(h)$ and $F_{\text{neg}}(h)$ meet:

$$Rel = 1 - \frac{(1 - F_{\text{pos}}(h')) + F_{\text{neg}}(h')}{2}. \tag{4}$$

The relationship between the reliability of annotations and the cumulative distribution functions of the positive and negative annotations is shown in Figure 1. If many positively annotated samples occur in the higher ranks, the value of $1 - F_{\text{pos}}(h)$ drops suddenly. In this case, the reliability Rel will have a large value (indicated by the arrow in Figure 1). In an opposite manner, if positively annotated samples are equally distributed in the ranking, $1 - F_{\text{pos}}(h)$ has a gentle slope. In this case, the reliability Rel will have a small value. That is, according to this criterion, the wider the separation between positive and negative annotations, the higher the reliability of annotations. The details of how the reliability varies according to the number of iterations is explained in the section on our experiments (Section 5).

4.3. Elimination of Incorrect Tags and Re-annotation

The procedure for the proposed method is shown in Figure 2. The way we input the training data (a set of image feature vectors and labels) and output a recognition model is the same as in general approaches, with the difference that our method includes iterative processes for the elimination of unreliable annotations (steps 4 to 14) and their re-annotation (steps 17 to 19). By repeating these two processes on a training image database that has noisy annotations, we can eliminate and modify irrelevant annotations and create a refined recognition model.

In the procedure for eliminating unreliable annotations (steps 4 to 14), we utilize $1 - F_{\text{pos}}(h)$ and $F_{\text{neg}}(h)$ again.

Specifically, for all the training image data X, if an image sample $\boldsymbol{x}_i$ (its rank is h-th) has

$$1 - F_{\mathrm{pos}}(h) \geq F_{\mathrm{neg}}(h) \quad \text{and} \quad l_i = negative \tag{5}$$

or

$$1 - F_{\mathrm{pos}}(h) < F_{\mathrm{neg}}(h) \quad \text{and} \quad l_i = positive, \tag{6}$$

we temporarily delete the annotation by setting $l_i = unknown$. The data in Equation (5) are at a higher rank and tagged as negative, and these are more likely to be incorrectly tagged. Likewise, the data in Equation (6) are at a lower rank and tagged as positive, and these are more likely to be incorrectly tagged as well. We temporarily delete the annotations for these data.

Next, we create a new recognition model using a training database that excludes the image samples with $l_i = unknown$, and then we re-evaluate the samples with $l_i = unknown$. When the score s_i is a positive value, it is more likely to be a positive sample; hence, we re-annotate it as $l_i = positive$. Similarly, when the score s_i is a negative value, we re-annotate it as $l_i = negative$.

5. Experiments

5.1. Image Database Used in Our Experiments

We conducted experiments using the Caltech 101 data set [23], which is generally used for generic object recognition. The Caltech 101 data set includes 101 categories of object images, with 31 to 800 images per category. The total number of images is 8,677. We randomly selected 30 images per category as training samples in the experiment. That is, 30 positive samples and 3,000 negative samples (30 images × 100 categories) were used as the training samples for each category, and the rest of the 5,647 (= 8,677 - 3,030) samples were used for the evaluation.

5.2. Performance Evaluation under No Annotation Errors

First, we evaluated the recognition performance for the case where there were no annotation errors in the training set.

For the feature extraction, we used a CNN proposed in ILSVRC 2012, specifically AlexNet [1]. After inputting an image to the network, we extracted a 4,096-dimensional feature vector from the hidden layer (the sixth layer) and used it as a visual feature.

For the recognition model, we used support vector machines (SVMs) with radial basis function (RBF) kernel. Using 30 positive samples and 3,000 negative samples for each category, we created a recognition model on the 4,096-dimensional feature space.

After we evaluated all the test samples and created a ranked list sorted by predicted scores, the mean of each category's average precision (mAP) was calculated as shown in Subsection 3.2. Our results show that the mAP was relatively high (89.1%) under the condition with no annotation errors.

TABLE 1. ERROR TYPES USED IN OUR EXPERIMENTS.

	The # of data annotated as positive		The # of data annotated as negative	
	The # of positive samples	The # of negative samples	The # of positive samples	The # of negative samples
Error type 1	30		3,000	
	9	21	21	**2,979**
Error type 2	75		2,955	
	15	60	15	**2,940**

5.3. Performance Evaluation under Many Annotation Errors

We created very challenging conditions for constructing an accurate recognition model by intentionally changing annotations of training samples from positive to negative and vice versa. For this experiment, we created two error types (error types 1 and 2) as shown in Table 1 and evaluated the performance for each.

For error type 1, 21 of the 30 positive annotations were deliberately exchanged with 21 of the 3,000 negative annotations. This is a very difficult condition because only 9 positive samples (30% of the positive samples) were regarded as positive and 70% of the positive samples were regarded as negative.

Meanwhile, for error type 2, 15 of the 30 positive samples were regarded as negative and 60 of the 3,000 negative samples were regarded as positive. This is also a very challenging setting.

Under these conditions, we evaluated the performance using the same method as in Subsection 5.2. We found that the mAPs drastically deteriorated: from 89.1% to 35.5% for error type 1 and to 49.9% for error type 2.

5.4. Confirmation of the Effectiveness of Annotation Refinement

In order to confirm the effectiveness of our annotation refinement, we created a recognition model by following the algorithm described in Figure 2 and evaluated its performance.

We explain the details of the experiments below. In steps 5 to 7 in Figure 2, we calculated scores by 2-fold cross validation more than once ($K = 2, P = 4$ in our experiments) and took their average to reduce the impact of biased training samples and acquire more reliable scores. For the feature extraction in the four trials, in order to improve the robustness, we extracted features using four different CNN layers: the sixth and the seventh layers of the CNN that was trained with the ImageNet database [1], and the sixth and the seventh layers of the CNN called Hybrid-AlexNet [24], which was trained with data combined from the Places database (scene database) and the ImageNet database.

In steps 10 to 13, the iterative process is stopped based on the reliability of annotations Rel_r. That is, as long as the model's reliability increases, we repeat the annotation

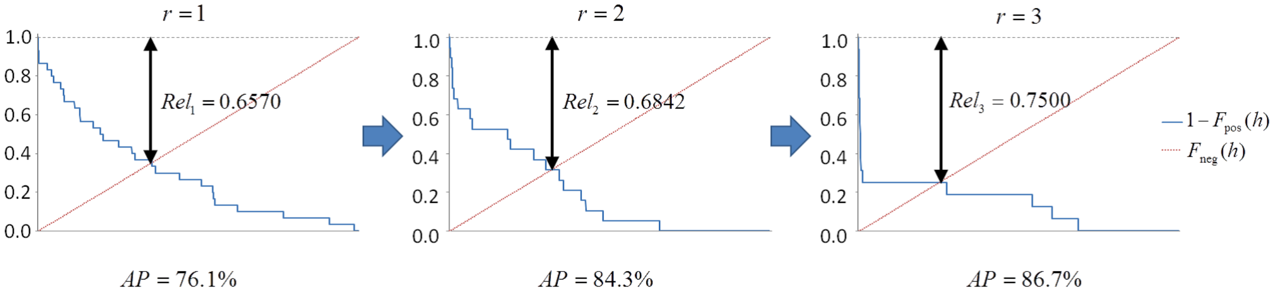

Figure 3. The progress of reliabilities for the category `umbrella`. (The error type is 1.)

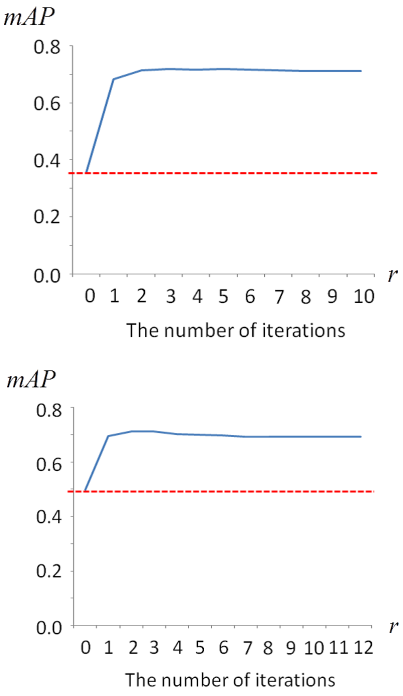

Figure 4. The relationship between the number of iterations and mAP. Top: error type 1. Bottom: error type 2. Red dotted lines show the performance under no annotation refinement (the baseline method).

refinement process. From the examples shown in Figure 3, it can be seen that reliabilities of annotations increased (Rel_r = 0.6570, 0.6842, 0.7500) with an increase in the number of iterations. This is because the number of positively annotated samples at a higher rank increases and the separation between positive and negative annotations becomes higher. At the same time, the performance of the recognition models improved as well (AP = 76.1%, 84.3%, 86.7%).

We created recognition models for all 101 categories and evaluated the mAPs. We found that the mAPs improved markedly, from 35.5% to 71.0% for error type 1 and from 49.9% to 69.2% for error type 2. Looking at the performance for each category, we found that 94 of the 101 categories were improved for error type 1 and 88 categories were improved for error type 2.

The relationship between the number of iterations and the mAP is shown in Figure 4. From this figure, we can see that the performance improved dramatically at the first iteration, improved further at the second iteration, and then converged at around the third iteration. A slight degradation can be seen after that. In a detailed investigation into the cause of this, we discovered that models for some categories began to recognize another specific category in the middle of the iterations. For example, for the category `wheelchair`, the annotation refinement process produced the intended effect in the beginning (iterations $r = 1$ to 3). However, the category `Motorbikes` gradually became dominant after iteration $r = 6$, and finally the majority of the top-ranked data were `Motorbikes` at the convergence point. Other categories tended to have similar kinds of errors. The reason is that the two categories are visually similar, such as `wheelchair` vs. `Motorbikes` and `Face` (with images that include background) vs. `Face_easy` (with images that have less background).

6. Summary and Future Work

In this paper, we have dealt with the problem of creating a high-performance recognition model under conditions of very noisy annotations. Our method iteratively refines annotations based on the reliability of annotations. Using an image database with many annotation errors, our experiments showed that the proposed method dramatically improved the image retrieval performance for most of the image object categories. One of the advantages of our method is that we do not need a parameter tuning, so this method can be used in actual Internet environments where the proportion of erroneous annotations is unknown.

Some large-scale image databases have now been developed, such as the ImageNet database, which contains several million images from thousands of categories. On the other hand, an incomparably greater number of images have been uploaded to the Internet. Future work, therefore, will be to collect massive amounts of image samples from the Internet and create a better image recognition model using our proposed refinement method.

Acknowledgements

This work was partially supported by JSPS KAKENHI Grant Number 15K00249 and Waseda University Grant for Special Research Projects 2016A-026.

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” Advances in Neural Information Processing Systems, vol.25, pp.1106–1114, 2012.

[2] D. Liu, X.-S. Hua, L. Yang, M. Wang, and H.-J. Zhang, “Tag ranking,” In Proc. of WWW, pp.351–360, 2009.

[3] X. Li, T. Uricchio, L. Ballan, M. Bertini, C. G. M. Snoek, and A. D. Bimbo, “Socializing the semantic gap: A comparative survey on image tag assignment, refinement and retrieval,” arXiv:1503.08248 , 2015.

[4] A. Makadia, V. Pavlovic, and S. Kumar, “Baselines for image annotation,” International Journal of Computer Vision, vol.90, no.1 pp.88–105, 2010.

[5] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid, “TagProp: Discriminative metric learning in nearest neighbor models for image auto-annotation,” In Proc. of ICCV, pp.309–316, 2009.

[6] J. Verbeek, M. Guillaumin, T. Mensink, and C. Schmid, “Image annotation with TagProp on the MIRFLICKR set,” In Proc. of ACM MIR, pp.537–546, 2010.

[7] J. Tang, R. Hong, S. Yan, T.-S. Chua, G.-J. Qi, and R. Jain, “Image annotation by kNN-sparse graph-based label propagation over noisily tagged web images,” ACM Transactions on Intelligent Systems and Technology, vol.2, no.2 pp.1–15, 2011.

[8] D. Liu, X.-S. Hua, M. Wang, and H.-J. Zhang, “Image retagging,” In Proc. of ACM MM, pp.491–500, 2010.

[9] L. Wu, R. Jin, and A. Jain, “Tag completion for image retrieval,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.35, no.3, pp.716–727, 2013.

[10] A. Znaidia, H. Le Borgne, and C. Hudelot, “Tag completion based on belief theory and neighbor voting,” In Proc. of ACM ICMR, pp.49–56, 2013.

[11] Z. Lin, G. Ding, M. Hu, J. Wang, and X. Ye, “Image tag completion via image-specific and tag-specific linear sparse reconstructions,” In Proc. of CVPR, pp.1618–1625, 2013.

[12] Z. Feng, S. Feng, R. Jin, and A. Jain, “Image tag completion by noisy matrix recovery,” In Proc. of ECCV, pp.424–438, 2014.

[13] X. Li, C. Snoek, and M. Worring, “Learning social tag relevance by neighbor voting,” IEEE Transactions on Multimedia vol.11, no.7, pp.1310–1322, 2009.

[14] L. Duan, W. Li, I. Tsang, and D. Xu, “Improving web image search by bag-based reranking,” IEEE Transactions on Image Processing, vol.20, no.11, pp.3280–3290, 2011.

[15] A. Sun, S. Bhowmick, K. Nguyen, and G. Bai, “Tag-based social image retrieval: An empirical evaluation,” Journal of the American Society for Information Science and Technology, vol.62, no.12, pp.2364–2381, 2011.

[16] Y. Gao, M. Wang, Z.-J. Zha, J. Shen, X. Li, and X. Wu, “Visual-textual joint relevance learning for tag-based social image search,” IEEE Transactions on Image Processing, vol.22, no.1, pp.363–376, 2013.

[17] B. Sigurbjörnsson and R. van Zwol, “Flickr tag recommendation based on collective knowledge,” In Proc. of WWW, pp.327–336, 2008.

[18] S. Zhu, C.-W. Ngo, and Y.-G. Jiang, “Sampling and ontologically pooling web images for visual concept learning,” IEEE Transactions on Multimedia vol.14, no.4, pp.1068–1078, 2012.

[19] G. Zhu, S. Yan, and Y. Ma, “Image tag refinement towards low-rank, content-tag prior and error sparsity,” In Proc. of ACM MM. pp.461–470, 2010.

[20] L. Chen, D. Xu, I. Tsang, and J. Luo, “Tag-based image retrieval improved by augmented features and group-based refinement,” IEEE Transactions on Multimedia vol.14, no.4, pp.1057–1067, 2012.

[21] X. Li and C. Snoek, “Classifying tag relevance with relevant positive and negative examples,” In Proc. of ACM MM. pp.485–488, 2013.

[22] J. Sang, C. Xu, and J. Liu, User-aware image tag refinement via ternary semantic analysis,” IEEE Transactions on Multimedia vol.14, no.3, pp.883–895, 2012.

[23] L. Fei-Fei, R. Fergus, and P. Perona, “Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories,” Computer Vision and Image Understanding, vol.106, no.1, pp.59–70, 2007.

[24] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, “Learning deep features for scene recognition using places database,” In Neural Information Processing System, vol.27, pp.487–495, 2014.