IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 5, 2020, accepted April 16, 2020, date of publication April 21, 2020, date of current version May 6, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2989200

A Weighted Topic Model Learned From Local Semantic Space for Automatic Image Annotation

HAIYU SONG[1,2], PENGJIE WANG[1], JIAN YUN[1], WEI LI[1,2], BO XUE[1], AND GANG WU[1]
[1]School of Computer Science and Engineering, Dalian Nationalities University, Dalian 116600, China
[2]College of Traffic, Jilin University, Changchun 130022, China

Corresponding author: Jian Yun (yunjianm@163.com)

This work was supported in part by the Liaoning Natural Science Foundation Projects under Grant 2019-ZD-0182, and in part by the Creative Research Talents from Universities of Liaoning Province under Grant LR2016071.

ABSTRACT Automatic image annotation plays a significant role in image understanding, retrieval, classification, and indexing. Today, it is becoming increasingly important in order to annotate large-scale social media images from content-sharing websites and social networks. These social images are usually annotated by user-provided low-quality tags. The topic model is considered as a promising method to describe these weak-labeling images by learning latent representations of training samples. The recent annotation methods based on topic models have two shortcomings. First, they are difficult to scale to a large-scale image dataset. Second, they can not be used to online image repository because of continuous addition of new images and new tags. In this paper, we propose a novel annotation method based on topic model, namely local learning-based probabilistic latent semantic analysis (LL-PLSA), to solve the above problems. The key idea is to train a weighted topic model for a given test image on its semantic neighborhood consisting of a fixed number of semantically and visually similar images. This method can scale to a large-scale image database, as training samples involved in modeling are a few nearest neighbors rather than the entire database. Moreover, this proposed topic model, online customized for the test image, naturally addresses the issue of continuous addition of new images and new tags in a database. Extensive experiments on three benchmark datasets demonstrate that the proposed method significantly outperforms the state-of-the-art especially in terms of overall metrics.

INDEX TERMS Automatic image annotation, image retrieval, probabilistic latent semantic analysis, topic model.

I. INTRODUCTION

Automatic image annotation (AIA) is an important and challenging task in the field of computer vision. The AIA techniques attempt to learn a model from the training images and predict semantic labels for test images automatically using the learned model. The outburst of multimedia content on the Internet as well as social media has raised the demands for automatic annotation methods, thus making it an active area of research [1]. In recent years, it has been an active research topic due to its great potential applications in image retrieval, image classification, and image understanding (or image caption generation) [1]–[3].

In the past 20 years, some representative AIA approaches have been proposed and great achievements have been made, such as MBRM [4], JEC [5], and 2PKNN [1]. These state-of-the-art AIA approaches can achieve better performance in the ideal image databases created by experts. However, they are not suitable for the realistic social images from content-sharing websites (e.g. Picassa, Flickr, and YouTube) and social networks (e.g. Facebook and Wechat). Those social images have two obvious characteristics. The first is that those images are always weakly annotated with user-provided tags. Those databases contain hundreds of millions of images of all kinds of quality, size, and content that are collected from non-professional users. In contrast to traditionally well-annotated image databases by experts, user-provided tags from those databases usually are subjective, incomplete, containing noisy words. Even worse, some images do not have tags at all. The second is the continuous addition of new images and new labels, which will lead to frequent modeling on large-scale image databases.

To accurately describe weakly annotated image content, topic models have been introduced for image representation

The associate editor coordinating the review of this manuscript and approving it for publication was Li He.

and understanding. In topic models, each image is represented as a mixture of latent topics, and each topic is described by a distribution over visual words [6]. The topic can be regarded as a means of image representation whose semantic level is higher than the visual feature. In comparison with visual word and textual word, the topic model is adapted to low-quality labels (synonym, polyseme, noisy, and incomplete tags) [7], [8]. The typical topic models introduced into computer vision are the latent semantic analysis (LSA), probabilistic latent semantic analysis (PLSA), and latent Dirichlet allocation (LDA) [6], [8]. For example, Li exploited PLSA and LDA for scene classification [9], [10]. Monay proposed the famous PLSA-WORDS based on PLSA for image annotation [7]. Tian proposed PLSA-MB based on PLSA for image annotation [11].

However, these state-of-the-art methods based on topic models have two problems. First, they are difficult to scale to a large-scale image dataset due to expensive storage cost or memory overhead, because the entire image dataset must be used to train a model. Second, because of continuous updating of the database and the subsequent model training, they can not be used to online image repository.

To overcome the above shortcomings, in this paper, we propose a novel annotation approach based on topic model, namely local learning-based PLSA (LL-PLSA), which aims to improve the semantic level, and reduce complexity of model training. Different from traditional topic models learned from the entire (training) database, our proposed model learns from a local semantic neighborhood consisting of only a small part of training images.

The main contributions of our work can be summarized as follows. (1) Novel image representation. We combine two different CNN features (visual and semantic) into our model to reduce semantic gap. (2) A weighted and customized topic model is proposed for image annotation. First, a weighting mechanism is introduced into topic model to improve the annotation model quality. Second, we propose the customized topic model learned online for each test image, instead of the same topic model learned offline for all images. (3) A fixed number of training images rather than all are used for training model. Experiment results on three annotation datasets (Corel5k [12], ESP Game [5], and IAPR TC-12 [5]) show that we only need 15 images for training a high-quality model. The experimental results and analysis demonstrate the effectiveness and efficiency of our approach.

II. RELATED WORKS

A. TOPIC MODEL APPROACHES

The probabilistic topic models are originally developed for natural language processing tasks [13]. They model a text document as a probabilistic distribution over some topics where each topic is itself defined as a probabilistic distribution over a set of words. In light of this, if we consider each patch of the image as a visual word and the whole image as a document consisting of these visual words, the topic model can be used for computer vision. In this model, both image patches (visual words) and annotation words (textual words) are assumed to be randomly generated conditioned on topic variables.

For an AIA approach based on standard topic model, all training images must be used to train a model in the training stage. As the number of model parameters grows linearly with the size of the image dataset, the storage overhead of model training is direct proportion to the size of training dataset. In a larger training dataset, the overhead of storage and time will dramatically increase, and even lead to memory overflow. Therefore, the traditional topic model can not be applied to a large-scale datasets. So, many researchers propose improved approaches. In order to improve image annotation or retrieval performance and reduce space-time overheads, there are two directions to improve the topic model, including extending model structure and decreasing training samples [14]–[16].

Some research works focus on extending model structures. Lienhart *et al.* [17] propose a multi-layer PLSA model that is convenient to extend the learning and inference rules to more layers and modalities. The MF-PLSA model [18] can be considered as an extension of PLSA methods in that it handles two kinds of visual feature domains. The tr-mmLDA model [19] presents a topic-regression multi-modal Latent Dirichlet Allocation method to learn the joint distribution of texts and image features. This tr-mmLDA model is quite different from the former topic models which only share a set of latent topics between two data modalities [8]. Song proposes a multi-modal topical coding approach for image annotation by capturing more compact correlations between textual words and image regions [15], [16].

Some research works aim to reduce training cost by selecting a subset of training image dataset. Romberg randomly selects some training images to train a PLSA model so as to reduce the storage overhead in the training model stage [20]. The improved PLSA model can effectively reduce storage overhead, but it suffers from model quality degrading due to discarding lots of valuable training images and semantic labels. Zheng classifies all images into different groups by SVM classifier, and train a PLSA model for each group [14]. Then, appropriate keywords are generated by label transfer algorithm in topic space. The proposed approach can improve the performance compared with standard PLSA. In terms of average precision, the annotation performance of the proposed approach is comparable to MBRM [14]. This improved topic model can reduce storage cost, but these SVM classifiers require to be retrained as new labels are added to the database. This method might not be applicable to online image repositories. Furthermore, in the large-scale image database, images of one group may be too many to train PLSA model. With regard to annotation performance, these annotation approaches based on PLSA might not achieve comparable performance, quoted as ''that alternative approaches based on probabilistic latent semantic indexing, while proposing appealing venues for linking the occurrences

of words and images, have not resulted in significant performance gains" [5].

B. NEAREST NEIGHBOR MODEL BASED APPROACHES

Recently, many nearest-neighbor models based AIA methods have been proved to be quite successful [8]. The Joint Equal Contribution (JEC) model is one of the most classical nearest-neighbor models [5]. The JEC model utilizes various low-level image features and a simple combination of basic distance measures to find nearest neighbors of a given image. It creates a family of very simple and intuitive baseline method for image annotation.

The two-pass KNN (2PKNN) model represents a classical solution to solve problems related to label-imbalance and weak-labeling [1]. It identifies all related semantic neighbors for each label by selecting K similar images in the vocabulary to boost rare labels. Due to its successfully solving class-imbalance problem, the 2PKNN makes great achievement in terms of popular evaluation metrics including per-word precision and recall and still is one of the most famous and influential image annotation approaches [8].

C. DEEP LEARNING BASED APPROACHES

Recently, Convolutional Neural Networks (CNNs) have shown great performance in many computer vision tasks by extracting effective feature vectors from images [21]–[25]. The approaches of image annotation based on deep learning can be classified into two categories. In the first category, the deep learning networks can perform image annotation by the means of multi-label multi-class classification. In this case, most approaches focus on modifying the output layer or activation function based on standard deep learning architecture. To adapt to a target image dataset, these modified models always require to be trained on a large image dataset, or fine-tuned on the target image dataset. In the second category, the function of deep learning models is just to extract feature vector from image. Most of the deep learning-based AIA approaches are based on convolution neural network (CNN) [24]–[27], and the features are always extracted from pre-trained AlexNet and VGGNet network [16], [25]–[27].

Jia, the creator of the Caffe, proposes CNN+WARP model [24]. The network architecture of the proposed model is similar to AlexNet. The experiments conducted on NUS-WIDE dataset outperform the conventional visual features by about 10% [24]. The CNN-R model is proposed by the team led by Murthy *et al.* [25], who proposed the famous CMRM and MBRM models. Their idea is to formulate the image annotation problem as a CNN based regression. The proposed method is based on CNN feature and word embedding vectors (word2vec). They achieve this by replacing the last layer of Caffe-Net with a projection layer (fully connected layer) and they call it as a CNN regressor (CNN-R). Verma and Jawahar perform the 2PKNN experiment using CNN-features and establish a new state-of-the-art performance [1]. Different from the aforementioned annotation methods based on CNN model, the D^{2}IA approach is based on generative adversarial network (GAN) model. The D^{2}IA aim to create semantically relevant, yet distinct and diverse labels [28]. Deep learning models are extensively being used for various computer vision tasks and shown a breakthrough performance, which mainly contributes to end-to-end feature extraction through convolution neural networks, but the application of deep learning for AIA is still in its early stage [2]. The deep learning-based AIA is a quite new but promising direction for AIA [8].

III. PROPOSED APPROACH

A. THE AIA FRAMEWORK

Figure 1 shows our proposed LL-PLSA framework for automatic image annotation. We extract two levels of feature vectors, the shallow layer visual feature and the deeper layer semantic feature, from the same deep convolution neural network (DCNN) (i.e. VGG-16). The semantic features aim to determine semantic neighbors of the test image, while visual features encoded as Bag-of-Words (BOW) aim to train the LL-PLSA topic model. After we get the neighborhood of the test image, we can obtain its corresponding sparse visual features (BOW). Thus, we build the local feature space, consisting of a few number of images semantically similar to the test image, where our proposed LL-PLSA is learned. The weight of each training sample for modeling is determined by the visual similarity between this training image and the test image. In other words, the topic model is directly learned from visual feature space constituted by neighbors, but the neighborhood is generated according to semantic features. Consequently, both visual and semantic features of images participate in our modeling, even though we never use any textual information. Different from traditional PLSA modeling, in our modeling process, the test image is involved in our modeling process, including determining neighbors and their weights. The topic model is customized for each test image online. Finally, the LL-PLSA predicts annotation words for the test image.

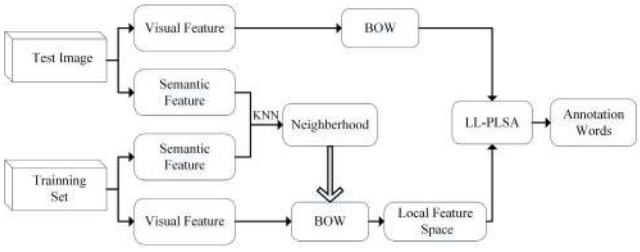


FIGURE 1. The proposed LL-PLSA framework.

B. THE SEMANTIC AND VISUAL FEATURES EXTRACTION BASED ON DEEP LEARNING

Recently, many developments in computer vision have been boosted by the use of DCNN, which achieves excellent performance especially for classification tasks [13]. Most DCNN architectures consist of convolution layers, max-pooling

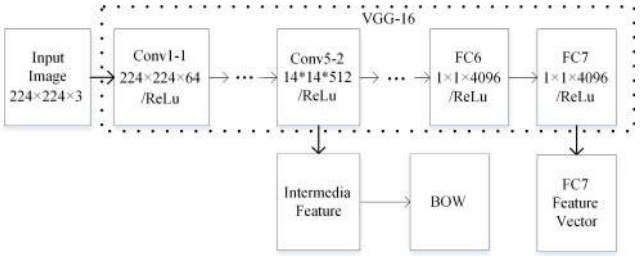


FIGURE 2. The flowchart of feature extraction.

layers, and fully-connected layers. In Figure 2, we illustrate VGG-16, a widely used DCNN in computer vision. VGG-16 includes thirteen convolution layers, three fully-connected layers, and five max-pooling layers. The convolution kernel of every convolution layer is 3×3, and every convolution layer is followed by ReLu activation function. As shown in Figure 2, the shape of each layer is represented by $R \times C \times K$. For example, the shape of the first convolution layer (namely Conv1-1) is $224 \times 224 \times 64$, which means that this layer has 64 feature maps and each has size 224×224.

In the architecture of DCNN, the deeper layers usually contain the semantic features, whereas shallower layers have detail features, namely visual features in this paper. To get semantic features, most works only utilize the output of the fully-connected layer while ignoring the output of intermediate convolution layers. In fact, as shown in Figure 2, the shallow layer (i.e. Conv5-2) can output a low dimension (512D) vector for each patch, whereas the deeper fully-connected layer (FC6-FC7) can output a high dimension (4096D) vector for an image. In Conv5-2 layer, an image includes 196 (14×14) patches, as a result, the Conv5-2 layer can output 196 vectors for an image. In other words, one FC vector represents a whole image, whereas one convolution vector only represents a patch of the image. As a result, the shallow layers capture more local visual patterns of objects, whereas deeper layers abstract more categorical information of the whole image.

Image classification and annotation need different ways of describing the image. Image classification aims to identify labels for an image from among a group of predetermined labels, whereas image annotation focuses on describing the objects in an image by a set of related keywords [13]. The class label can be considered as the global descriptor of the image, whereas the annotation words can be considered as the local descriptors of the objects in the image. To benefit from both semantic and visual features, we extract multi-level features from different layers simultaneously for AIA.

We have performed experiments on classical datasets (Corel5k, ESP Game, and IAPR TC-12) to verify the performance of each convolution layer in VGG-16 pre-trained on ImageNet2012 database. The Conv5-2 and Conv5-3 can achieve similar performance, both layers outperforming other convolution layers. These layers are the same both in the number of feature maps (196) and vector dimension (512D). We select the convolution output of the Conv5-2 as our intermediate feature. Then, we encode the output vectors of this intermediate (Conv5-2) using BOW coding. Specifically, to aggregate BOW feature, the codebook (vocabulary) is constructed from the training dataset in an unsupervised fashion (i.e. k-means). Using this codebook, each feature vector of patch is encoded. At last, an image is visually represented by a BOW feature vector. In Figure 2, the BOW feature vector is our visual feature, whereas the FC7 feature vector is our semantic feature vector.

C. WEIGHTED PLSA MODEL FOR IMAGE ANNOTATION

Typical probabilistic topic models include PLSA, LDA, and their variations. The PLSA is a general probabilistic topic model, whereas LDA is a special form of PLSA. The only difference is that the topic distribution of LDA is assumed to have a Dirichlet prior. PLSA may be equivalent to the LDA model under a uniform Dirichlet prior distribution. For simplicity but without loss of generality, we use PLSA as the following topic model.

The PLSA model assumes the existence of a latent topic z (aspect) in the generative process of each element x_j in a particular document d_i. The model can be defined by the joint probability of element x_j and document d_i as Equation (1),

$$P(x_j, d_i) = P(d_i) \sum_{k=1}^{N_z} P(z_k|d_i)P(x_j|z_k). \tag{1}$$

Equation (1) is equivalent to Equation (2),

$$P(x_j|d_i) = \sum_{k=1}^{N_z} P(z_k|d_i)P(x_j|z_k). \tag{2}$$

where N_z is the topic number. The parameters of the model are $P(z_k|d_i)$ and $P(x_j|z_k)$ that can be learned from training samples via likelihood maximization. To maximize the predictive probability $P(x_j,d_i)$, these parameters can be solved by using the Expectation-Maximization (EM) algorithm for the log-likelihood:

$$L = \sum_{i=1}^{N_d} \sum_{j=1}^{N_x} n(x_j, d_i) \log P(x_j, d_i). \tag{3}$$

where observation pair $n(x_j, d_i)$ denotes the number of element x_j in document d_i, N_d is the number of the documents, N_x is the number of the elements. The EM algorithm can be divided into two steps (E-step and M-step), which is defined respectively as,

E-step:

$$P(z_k|d_i, x_j) = \frac{P(x_j|z_k)P(z_k|d_i)}{\sum_{k=1}^{N_z} P(x_j|z_k)P(z_k|d_i)}. \tag{4}$$

M-step:

$$P(x_j|z_k) = \frac{\sum_{i=1}^{N_d} n(x_j, d_i)P(z_k|d_i, x_j)}{\sum_{j=1}^{N_x} \sum_{i=1}^{N_d} n(x_j, d_i)P(z_k|d_i, x_j)}. \tag{5}$$

$$P(z_k|d_i) = \frac{\sum_{j=1}^{N_x} n(x_j, d_i)P(z_k|d_i, x_j)}{\sum_{j=1}^{N_x} n(x_j, d_i)}. \tag{6}$$

In the process of model learning, the observation objects are the co-occurrence matrices between all samples (training images) and their elements rather than one sample and its elements. As a result, the probability tables in Equation (2), including $P(x|d)$, $P(z|d)$ and $P(x|z)$, will be large matrices. For $P(x|d)$, its row is as many as the number of images (N_d), whereas its column is as many as elements number (N_x). In a larger training dataset, the model will bring more overhead in both time and space. The model can't be applied to online image repositories due to memory overflow.

We propose a weighted PLSA model to further improve the model. Specifically, we modify the PLSA model equations by introducing weights to measure relevance between the training images and the test image. This improvement is based on the fact that the contribution of each training sample to the model should be proportional to how it is similar to the test image, instead of equal contribution. The weight of training sample d is defined by the similarity between the training sample d and test image d_{test}, which is embodied by weighted observation pairs of document d and element x in the process of modeling. The weighted observation pairs of document d and element x is defined as,

$$n_{wt}(x, d) = n(x, d) \times sim(d, d_{test}). \tag{7}$$

where $sim(d, d_{test})$ is the similarity between the test image d_{test} and training sample d. In LL-PLSA model, we replace the observation pars $n(x, d)$ with the weighted observation pairs $n_{wt}(x, d)$. Accordingly, we redefine Equation (3), (5), (6) respectively as,

$$L = \sum_{i=1}^{N_d} \sum_{j=1}^{N_x} n_{wt}(x_j, d_i) \log P(x_j, d_i). \tag{8}$$

$$P(x_j|z_k) = \frac{\sum_{i=1}^{N_d} n_{wt}(x_j, d_i) P(z_k|d_i, x_j)}{\sum_{j=1}^{N_x} \sum_{i=1}^{N_d} n_{wt}(x_j, d_i) P(z_k|d_i, x_j)}. \tag{9}$$

$$P(z_k|d_i) = \frac{\sum_{j=1}^{N_x} n_{wt}(x_j, d_i) P(z_k|d_i, x_j)}{\sum_{j=1}^{N_x} n_{wt}(x_j, d_i)}. \tag{10}$$

where, element x can represent an annotation word t (in textual modality) or visual word v (in visual modality), $n_{wt}(x_j, d_i)$ is weighted representation of the observed frequencies of neighbor image d_i and element x_j.

In the proposed local learning-based PLSA (LL-PLSA), d represents neighbor image of the test image instead of all training images. N_d, N_x and N_z are only associated with a fixed number of neighbor images instead of the entire training dataset. Hence, we can dramatically reduce the time and space overheads. Given a test image, its topic model can be learned from the semantic neighborhood. The neighbor images are a fixed number of training images that are most semantically similar to the test image according to their semantic feature vectors (FC7). The generation details of the neighborhood and semantic feature vector are described in section 3.1. Even in large-scale online image repositories, our proposed approach only needs about a few number of semantic neighbor images to train a model. The storage and time costs of model training are a small constant if ignoring the cost of generating the semantic neighborhood.

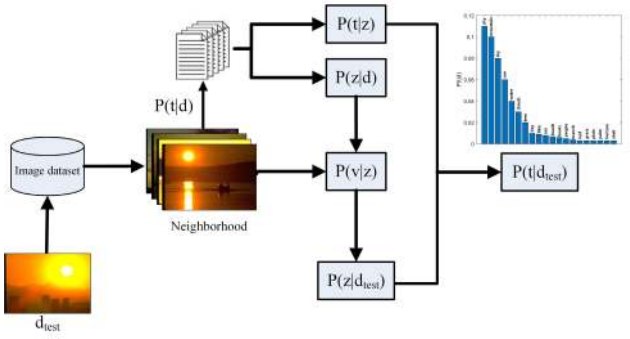


FIGURE 3. The flowchart of LL-PLSA modeling and annotation.

The flowchart of LL-PLSA modeling and annotation is shown as Figure 3. Given a test image d_{test}, its neighborhood can be generated using semantic feature similarities between test image and training dataset. The model learning and annotation are as follows. Firstly, a textual topic model can be trained on textual modality of neighbor images to learn both $P(z|d)$ and $P(t|z)$ parameters according to Equations (1),(4),(8)-(10). Secondly, a visual topic model can be trained on visual modality to learn $P(v|z)$ with fixed $P(z|d)$ which is learned previously in textual topic model. Thirdly, a visual topic model for the test image can be trained to learn $P(z|d_{test})$ while the previously learned $P(v|z)$ is fixed. Finally, for given the test image, we can infer the posterior probabilities of annotation words by the following equation

$$P(t|d_{test}) = \sum_{k=1}^{N_z} P(t|z_k) P(z_k|d_{test}). \tag{11}$$

The mode training process and annotation process of the proposed model can be summarized as the Algorithm 1 and Algorithm 2, respectively.

IV. EXPERIMENT

A. DATASET

We perform our experiments on three datasets including Corel5k, ESP Game and IAPR TC-12. The images in these datasets are of various categories such as natural scene, game, sketches, personal photos and so on, which makes the annotation a challenging task. Corel5K was the first widely used in 2002, and since then it has become a de facto evaluation benchmark for comparing the annotation performance [1], [12]. Corel5k dataset consists of 4500 training images and 499 test images. Each image is either 192 × 128 or 128 × 192 pixels. Each image is manually annotated with 3.5 words on average from a dictionary of 260 words. ESP Game dataset consists of 18689 training images and 2081 test images. Each image is manually annotated with 4.7 words on average from a dictionary of 268 words. IAPR TC-12 dataset consists of 17665 training images and 1962 test images. The images are 480 × 360 or 360 × 480 pixels. Each image

Algorithm 1 Training Model for a Given Test Image

Require: visual feature v of the test image, visual features v and textual words t of the test's neighbor images
//modeling in textual modality of semantic neighborhood
Process:
Randomly initialize the probability tables $P(z|d)$ and $P(t|z)$
while $\Delta L > T$ **do**
  for $k \in 1, \ldots, N_z$ and all (d_i, t_j) pairs in neighborhood **do**
    Compute $P(z_k|d_i, t_j)$ with Equation (4)
  end for
  for $j \in 1, \ldots, N_t$ and $k \in 1, \ldots, N_z$ **do**
    Update $P(t_j|z_k)$ with Equation (9)
  end for
  for $i \in 1, \ldots, N_d$ and $k \in 1, \ldots, N_z$ **do**
    Update $P(z_k|d_i)$ with Equation (10)
  end for
  Compute L from $P(t_j|z_k)$ and $P(z_k|d_i)$ with Equation (1) and (8)
end while
Save $\theta_t = \{P(t_j|z_k)\}$, $j \in 1, \ldots, N_t$, $k \in 1, \ldots, N_z$
//modeling in visual modality of semantic neighborhood
Randomly initialize the probability table $P(v|z)$
while $\Delta L > T$ **do**
  for $k \in 1, \ldots, N_z$ and all (d_i, v_j) pairs in neighborhood **do**
    Compute $P(z_k|d_i, v_j)$ with Equation (4)
  end for
  for $j \in 1, \ldots, N_v$ and $k \in 1, \ldots, N_z$ **do**
    Update $P(v_j|z_k)$ with Equation (9)
  end for
  Compute L from $P(v_j|z_k)$ and $P(z_k|d_i)$ learned in textual modality with Equation (1) and (8)
end while
Save $\theta_v = \{P(v_j|z_k)\}, j \in 1, \ldots, N_v, k \in 1, \ldots, N_z$
Ensure: θ_t and θ_v

Algorithm 2 Annotating the Test Image

Require: visual features v of the test image, θ_t and θ_v
Process:
//modeling in visual modality of test image with the parameter θ_v
Randomly initialize the vector $P(z|d_{test})$
while $\Delta L > T$ **do**
  for $k \in 1, \ldots, N_z$ and all (d_{test}, v_j) pairs **do**
    Compute $P(z_k|d_{test}, v_j)$ from $P(z_k|d_{test})$ and θ_v with Equation (4)
  end for
  for $k \in 1, \ldots, N_z$ **do**
    Update $P(z_k|d_{test})$ with Equation (10)
  end for
  Compute L from $P(z_k|d_{test})$ and θ_v with Equation (1) and (8)
end while
//annotating test image with the parameter θ_t
Compute $P(t|d_{test})$ from $P(z|d_{test})$ and θ_t with Equation (11)
Select top 5 words with the highest distributions of $P(t|d_{test})$ as annotation words
Ensure: Annotation words of the test image

is manually annotated with 5.7 words on average from a dictionary of 291 words.

Besides the above datasets, there are also other publicly available image datasets such as UIUC-Sports. However, these datasets have no specific divisions between training and test images. For a fair comparison, we will not collect results on these datasets.

B. EVALUATION METRICS

The precision, recall, and F1-measure are the most popular metrics in the information retrieval and AIA communities. Recently, per-word metrics (per-word precision, recall, and F1-measure) are most widely used in the AIA community. For each word, per-word precision is defined as the number of images correctly predicted over the total number of images predicted with this word, and per-word recall is defined as the number of images correctly predicted over the total number of images having this word in its ground-truth or manual annotations. The values are averaged over all the words in the vocabulary as,

$$\text{per-word Precision} = \frac{1}{C}\sum_{i=1}^{C}\frac{N_i^c}{N_i^p}. \tag{12}$$

$$\text{per-word Recall} = \frac{1}{C}\sum_{i=1}^{C}\frac{N_i^c}{N_i^g}. \tag{13}$$

where, C is the number of labels (keyword or classes), N_i^c is the number of images correctly annotated for label i, N_i^p is the number of predictions for label i, and N_i^g is the number of images having ground-truth label i.

Many researchers have pointed out that the per-word metrics are biased toward infrequent labels, because making them correct could have a very significant impact on final accuracy [24]. Therefore they propose overall metrics (sometimes called as per-image metrics) [24], [29]–[31]. The overall metrics are defined as,

$$\text{overall Precision} = \frac{\sum_{i=1}^{C} N_i^c}{\sum_{i=1}^{C} N_i^p}. \tag{14}$$

$$\text{overall Recall} = \frac{\sum_{i=1}^{C} N_i^c}{\sum_{i=1}^{C} N_i^g}. \tag{15}$$

The F1-measure is used for comprehensive performance evaluation by combing precision and recall. The same equation can be used for both per-word metric and overall metric

as,

$$\text{F1-measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \quad (16)$$

C. IMPLEMENTATION DETAILS

To conveniently compare different AIA models, we employ VGG-16 to collect visual and semantic features by removing the final fully-connected layer. We use FC7 of VGG-16 to extract 4096-dimensional semantic feature vector for our experiments. In the codebook generation step, the number of clusters (code words) is set to 1000. In our experiments, given an image from any dataset, we can generate its 1000-dimensional visual feature vector according to BOW model. We need to resize all images of the three datasets to 224×224 so that it is compatible with the input size of the VGG-16. It is worthy to note that the VGG-16 network used in this paper is pre-trained on ImageNet2012 dataset [22] without retraining or fine-tuning on our target dataset to demonstrate our model generality.

The similarity measure between two images is defined as the cosine similarity between their features. The number of nearest neighbors K is fixed to fifteen. As our proposed approach needs modeling for every test image, it is impossible to manually set topics number every time. We consider the number of the topics is related to the number of annotation words associated with training samples. We empirically set topics number as number of keywords existing in semantic neighbor images of the test image.

D. RESULTS AND COMPARISON

In this subsection, we evaluate our method and previous methods on three datasets (Corel5k, ESP Game and IAPR TC-12) using per-word metrics (precision, recall, and F1-measure) denoted as W-P, W-R, and W-F respectively. For a fair comparison, we carry out our experiment on the same datasets using DCNN and predict a fixed length of annotations (five words) for each test image.

In Table 1, we compare our approach with state-of-the-art models, including classical topic model PLSA-WORDS, the latest topic model PLSA-MB, classical probabilistic model MBRM, two nearest-neighbor models JEC and TagProp. We also compare with D^{2}IA that is a method based on GAN, a representative deep architecture. There are three kinds of features used in these methods including handcraft feature (HC), deep feature from VGG-16, and deep feature from GAN.

Table 1 gives a performance comparison in terms of per-word metrics. From the Table 1, we can see that our method outperforms previous topic model-based methods (PLSA-WORDS, PLSA-MB) by a large margin, although we achieve comparable results against methods that do not fall in topic model category. As can be seen from Table 1, some methods outperform LL-PLSA in terms of certain metric, but our proposed LL-PLSA can achieve comparable performance (on ESP Game) and best performance (on Corel5k and IAPR TC-12) in terms of comprehensive metric (per-word F1-measure). Moreover, our proposed LL-PLSA can outperform any method in terms of per-word F1-measure as neighborhood size is larger than 28 as shown in Figure 5(a) in the next subsection.

TABLE 1. Performance comparison in terms of per-word metrics.

Model	Feature	Corel5K			ESP Game			IAPR TC-12		
		W-P	W-R	W-F	W-P	W-R	W-F	W-P	W-R	W-F
MBRM	HC	24	25	25	18	19	19	24	23	24
JEC	HC	27	32	29	25	16	20	23	19	21
PLSA-WORD	HC	16	25	20	11	12	12	15	15	15
PLSA-MB	HC	26	30	28	-	-	-	-	-	-
TagProp(ML)	HC	31	37	33	49	20	28	48	25	33
2PKNN	HC	**40**	39	39	**51**	23	31	**49**	32	39
JEC	VGG-16	31	32	32	26	22	24	28	21	24
D^{2}IA	GAN	-	-	-	31	**49**	**36**	33	**45**	35
PLSA-WORD	VGG-16	20	30	24	19	22	21	22	25	24
LL-PLSA(our)	VGG-16	37	**48**	**42**	37	33	35	46	35	**40**

E. FURTHER EVALUATION

In this subsection, we will comprehensively compare LL-PLSA, the classical topic model PLSA-WORDS, the state-of-the-art approach 2PKNN, and three most famous representative models (MBRM, JEC, and TagProp). In recent years, 2PKNN is considered as a state-of-the-art approach, which is referenced by nearly all AIA works. The 2PKNN was proposed for the first time by Yashaswi in 2012 [32], then Venkatesh and R. Manmatha implemented it using DCNN features in 2015 [25]. Yashaswi re-implemented this work using three layers of features from DCNN based on DeCAF framework in 2017 [1]. As DeCAF framework is obsolete today, we substitute this model with VGG-16 implemented in PyTorch. We extract 9192 dimensional feature from its last three fully-connected layers according to [1]. For a fair comparision, in this subsection, all approaches but MBRM are performed using deep features extracted from the same architecture (VGG-16). Being limited to model itself, MBRM is performed using handcraft feature.

The goal of image annotation is to facilitate image retrieval, image classification, and image understanding [1]–[3], whereas per-word metrics are mainly for evaluating image retrieval. In essence, as far as image understanding is concerned, the overall metrics are much better than per-word metrics, since the overall metrics pay more attention to image content understanding in the perspective of each image. As a result, in common with per-word metrics, the overall metrics are considered as appropriate metrics for comparing image annotation methods [24], [29]–[31]. In addition, we also introduce hybrid F1-measure (called H-F1) combining per-word F1-measure and overall F1-measure with the harmonic mean [29]. In Table 2, we denote per-word metrics by W-P, W-R, and W-F, whereas overall metrics by O-P, O-R, and O-F. The H-F1 is denoted as H-F.

As shown in Table 2, our LL-PLSA significantly outperforms MBRM, JEC, and TagProp in both per-word and overall metrics as we expect. TagProp achieves better

TABLE 2. Comprehensive performance comparison in terms of per-word and overall metrics.

dataset	Model	W-P	W-R	W-F	O-P	O-R	O-F	H-F
Corel5k	MBRM	24	25	24.49	32	45	37.40	29.60
	JEC	31	32	31.49	41	58	48.12	38.07
	TagProp	31	40	34.97	44	61	50.96	41.48
	PLSA-WORDS	20	30	23.87	35	50	41.07	30.19
	2PKNN	**37**	46	41.22	37	54	44.37	42.73
	LL-PLSA	**37**	**48**	**41.84**	**46**	**65**	**53.96**	**47.14**
ESP Game	MBRM	18	19	18.49	25	28	26.42	21.75
	JEC	26	22	23.83	33	36	34.46	28.18
	TagProp	36	28	31.61	38	42	39.87	35.26
	PLSA-WORDS	19	22	20.06	25	27	25.68	22.13
	2PKNN	**50**	24	32.44	36	39	37.41	34.74
	LL-PLSA	37	**33**	**35.01**	**41**	**46**	**43.51**	**38.80**
IAPR TC-12	MBRM	24	23	23.49	29	27	27.96	25.53
	JEC	28	21	24.00	43	41	41.88	30.51
	TagProp	43	35	38.83	48	46	47.05	42.55
	PLSA-WORDS	22	25	23.83	33	31	31.85	27.26
	2PKNN	**49**	32	38.83	43	41	41.65	40.19
	LL-PLSA	46	**35**	**39.70**	**51**	**49**	**50.11**	**44.30**

performance in both per-word and overall metrics, which probably benefits from its sophisticated metric learning. But it is hard to apply to a large-scale image database because of time-consuming metric learning in training stage.

Our approach outperforms PLSA-WORDS in terms of both per-word and overall metrics. This might attribute to our novel semantic neighborhood and weighted topic model customized for each test image. Our approach is comparable to 2PKNN in terms of per-word metrics, which suggests our proposed semantic neighbors can build high-quality semantic space. In contrast to 2PKNN, our semantic neighborhood is generated using deep learning features rather than textual labels. For an image, 2PKNN consider that its nearest neighbors from a given label constitute its semantic neighborhood with respect to that label, whereas our approach directly selects nearest neighbors from the entire image dataset and only generates a neighborhood rather than as many as the labels. Moreover, the deep learning feature is extracted from pre-trained model rather than fine-tuned model, and our semantic neighborhood generation is irrelevant to textual labels, so our approach is able to address the issue of new labels addition. Even if the number of user-generated tags is tremendous, our approach is able to generate semantic neighborhood in the same way, considering that the number of semantic neighbors is fixed and irrelevant to the number of labels. Therefore, our approach can be applied to both large-scale image datasets and online image repositories.

Our approach significantly outperforms 2PKNN in terms of overall metrics and H-F1. The main reason is that our performance improvement largely benefits from accurate image understanding whereas 2PKNN mainly from giving more importance to rare labels than the frequent ones. The infrequency labels are usually related to fewer images, whereas frequent ones related to more images. Enhancing infrequency labels implicitly sacrifices many images with frequency labels, which will decrease per-image performance of image understanding. Our topic model customized for a given test image is learned from its semantic neighbors, then the image is represented by the customized topic model. The topic feature can accurately capture more implicit information than the visual feature, which is in favor of image understanding and improving our performance in terms of overall metrics. In summary, our proposed method improves performance in both per-word and overall metrics, which results from accurate image understanding.

As the annotation performances are affected by the number of annotation words, we carry out evaluation experiments with the number of annotation words varying from 2 to 20. The overall precision-recall curves of MBRM, JEC, TagProp, PLSA-WORDS, 2PKNN, and our method are visualized in Figure 4 based on Corel5k, ESP Game, and IAPR TC-12, respectively. The overall precision and recall values are the mean values calculated over all the test images. As can be seen from Figure 4, our model remarkably outperforms the others for any number of annotation words. These again confirm the effectiveness and efficiency of our method.

F. QUALITATIVE ANALYSIS

Table 3 shows several examples of annotations produced by our approach on the three datasets. The example images in three rows are from Corel5k, ESP Game, and IAPR TC-12, respectively. We can see that our approach can correctly predicate the ground-truth annotations, although there are some extra labels. By checking the extra labels (with blue font), we find that most of them are all consistent with the content of the images but not included in ground-truth labels. Our approach can integrate visual information into the neighbor semantic space so as to find visually and semantically similar images and predicate the correct annotation words.

G. EFFICIENCY ANALYSIS

In this subsection, we analyze the efficiency of the proposed approach with varying neighborhood size and compare it with PLSA-WORDS. The experiments are performed using Matlab programs on a computer of Intel Core(TM) i5-6500 CPU with 3.20GHz and 16 GB RAM, running Windows 7 OS. We conduct evaluation experiments with the number of nearest neighbors K varying from 1 to 100, and show the influence of K on the annotation performance on all the three datasets (in terms of per-word F1-measure and overall F1-measure) in Figure 5(a) 5(b). We can observe the performances are generally stable but have a little fluctuation as K increases. The reason is that topic models are initialized randomly. In Figure 5(c), we analyze the computational costs of LL-PLSA on K varying from 1 to 100. Since the proposed model is customized for each test image, the training and testing process work alternately. To accurately compute time costs, we give the total time costs for all test images instead of one test image. Consequently, the time costs in Figure 5(c) is the total time of N training stages and N testing stages (N is the number of test images). The complexity of the proposed

TABLE 3. Example results on corel5k, ESP-game and IAPR TC-12 and the annotation predicted by our approaches.

Test Image	Ground-truth	Annotation	Test Image	Ground-truth	Annotation	Test Image	Ground-truth	Annotation
	Mountain sky water clouds	sky mountain water **tree** clouds		water sea boats display	water display **beach** sea boats		sky buildings fence flight	fence sky buildings **plane** flight
	dress girl party smile woman	girl woman party smile dress		couple man	man **woman** couple **suit** **smile**		red woman	**hair** **girl** woman **blonde** red
	bag jean shop tee-shirt woman	tee-shirt woman bag jean shop		building hill plant	plant hill building **tree** **house**		mountain slope	slope **people** mountain **rock** **man**

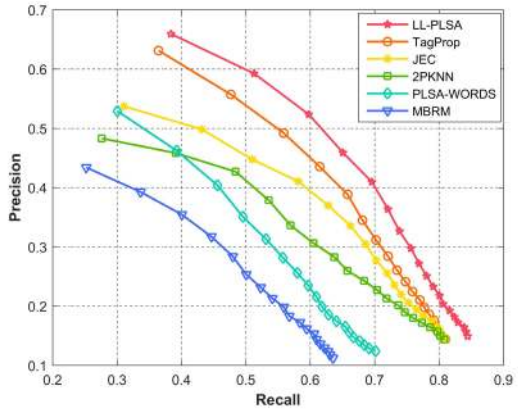
LL-PLSA model depends on the neighborhood instead of the whole dataset. As we expect, the time costs are almost linear for neighborhood sizes in the three datasets. We can observe that, for K larger than 15, the increase in annotation performance (both per-word and overall F1-measure) is very small as K increases, but the time costs will increase nearly linearly. Balancing efficiency and quality, we set K as 15.

To verify the efficiency of the proposed LL-PLSA, as shown in Table 4, we compare the time costs among PLSA-WORDS, 2PKNN, and LL-PLSA. In LL-PLSA experiments on the three datasets, the models are learned from fixed 15 neighbors. As shown in Table 4, the time cost of PLSA-WORDS includes one training stage and N testing stages, while that of LL-PLSA includes N training stages and N testing stages (N is the number of test images). As shown in Table 4, for every dataset, the total time cost of the proposed LL-PLSA is even smaller than the testing cost of PLSA-WORDS. A possible explanation is that the number of LL-PLSA model parameters is obviously smaller than that of PLSA-WORDS. For both LL-PLSA and PLSA-WORDS, theoretically, they need to learn the same probability tables, including $P(t|z)$, $P(z|d)$, $P(v|z)$ in the modeling stage, and $P(z|d_{test})$ in the testing stage. In fact, the number of LL-PLSA model parameters dramatically decrease compared with PLSA-WORDS. Taking $P(z|d)$ of Corel5k as an example, we compare the two models. For PLSA-WORDS, the numbers of training samples and topics, N and K, are 4500 and 200 respectively, while N and K are 15 and about 18 in LL-PLSA respectively. In the modeling process, $P(z|d)$ is a K-by-N table that stores model parameters of the N multinomial distributions $P(z|d_i)$. Thus, the $P(z|d)$ table includes 900 thousand (200×4500) and 270 (18×15) values for PLSA-WORDS and LL-PLSA respectively. Similarly, the numbers of the other parameters of PLSA-WORDS are much larger than those of LL-PLSA. The PLSA-WORDS model is so complex that its testing time cost is larger than the total time cost of training and testing of LL-PLSA.

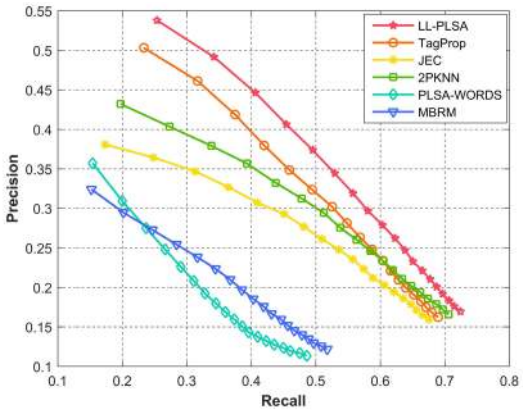
In contrast to original topic models, the complexity of the proposed LL-PLSA model is irrelevant to the size of training image database. Theoretically, for the same neighborhood size, the space-time overheads of LL-PLSA are almost fixed if ignoring differences in textual distribution between different databases. As can be seen from Table 4, for each dataset, the total time cost of LL-PLSA is nearly proportional to the number of test images. Moreover, its storage cost of modeling only depends on a fixed number of training sample images (including the numbers of images, textual words, visual words, and topics) rather than the entire training or test dataset. This again confirms that the proposed LL-PLSA can scale to any large-scale image database.

For most traditional AIA models based on nearest neighbors such as JEC, the performance will decrease as neighborhood size K increases. The most possible explanation is that more irrelevant training samples (with more weak and noisy labels) participate in modeling along with a larger neighborhood. It is worth mentioning that the performance of our proposed LL-PLSA is very stable as K increases as shown in Figure 5(a) 5(b), which mainly benefits from the topic model. In contrast to AIA models based on visual features, topic models can capture more abstract concepts and deal with low-quality manual labels. In comparison with classical PLSA models, our proposed method can achieve better performance at the low cost of storage and time as shown in Table 4. Hence, our approach will be more fit for online image repository with low-quality manual labels.

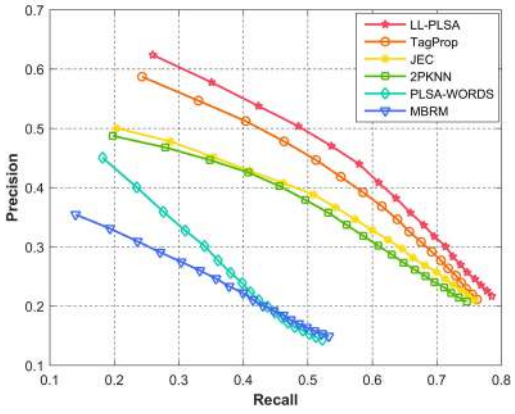
In contrast to traditional offline trained models, ours is online trained efficiently for a given test image, which ensures the current model consistent with the latest information of the real-world database. Hence, ours is capable of addressing the issue of continuous addition of new concepts and new



(a) Corel5k dataset



(b) ESP Game dataset



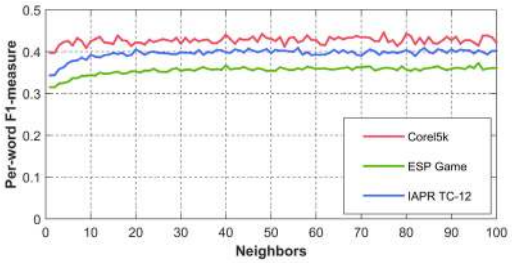
(c) IAPR TC-12dataset

FIGURE 4. Overall precision-recall curves on three datasets.

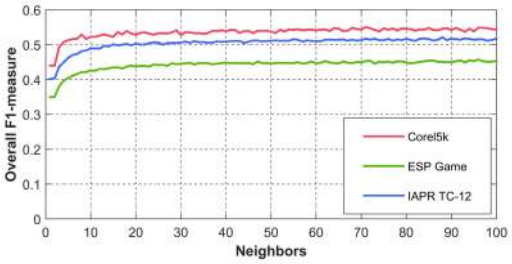
TABLE 4. Time costs of PLSA-WORDS, 2PKNN and LL-PLSA(in seconds).

	PLSA-WORDS		2PKNN	LL-PLSA
	Training	Testing	Training+Testing	Training+Testing
Corel5k	45.52	9.23	8.16	5.88
ESP Game	212.79	51.36	85.29	29.84
IAPR TC-12	480.37	43.75	80.54	30.23

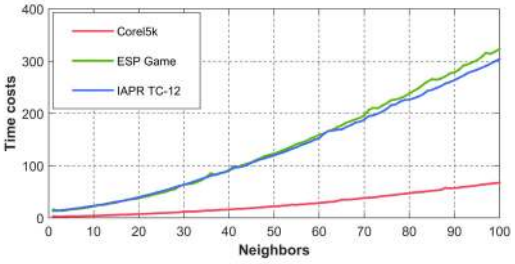
images in the real-world database. As mentioned above, even though the time of our model is shorter than the original topic model (PLSA-WORDS), our modeling procedure still seems



(a) Per-word F1-measure



(b) Overall F1-measure



(c) Time costs

FIGURE 5. Annotation effectiveness and efficiency of LL-PLSA with varying neighborhood size.

to be very time-consuming because the model needs to be trained for every test image. As nearest-neighbor methods are concept-clear and structure-intuitive, and they need not train complex models, to further verify the efficiency of LL-PLSA, we compare it with 2PKNN that is a two-step variant of the classical k-nearest neighbor algorithm. As shown in Table 4, the testing time of 2PKNN is much longer than the total time of training and testing of LL-PLSA. Therefore, despite our method needs to train a model for each test image, in fact, ours can perform annotation efficiently rather than suffer from the drawbacks of low generalization and customizing model for every test image. As a matter of fact, it is only online modeling that allows us to customize a model for the specific test image so as to apply to real-world image database.

V. CONCLUSION AND FUTURE WORK

We present a novel image annotation based on PLSA model. To our knowledge, this is the first published work that proposed to explicitly utilize the intermediate layer output of CNN as local visual feature in the image annotation task. Different from many works attempting to devise more complicated models, we propose the concept of customizing "agile" topic model for each test image, which is learned from local space consisting of a few number of visually and semantically

similar images. Our proposed LL-PLSA has several advantages. (1) Our LL-PLSA can easily fuse semantic feature and visual feature, extracted from the same deep learning architecture, into the semantically and visually meaningful neighborhood so as to train a low-cost high-quality topic model. (2) Our method can scale to a large-scale dataset because our neighborhood is a fixed number of semantic neighbor images (semantic neighborhood) instead of the entire training dataset. (3) Our method effectively exploits weighting mechanism into topic model to improve modeling quality so as to address the weak-labeling issue. (4) Our proposed weighted and customized topic model is fit to online image repository, whose characteristic is the continuous addition of new images and new labels. Extensive experiments demonstrate that our approach can achieve comparable or state-of-the-art performance in terms of per-word metrics, whereas our method significantly outperforms 2PKNN in terms of overall metrics. Our customized model for a given test image has many advantages, however, there is a shortcoming. The complete from-scratch model needs training for every test image, even though few tags and images are added to the image database. In the future, we will explore a new modeling strategy based on the existing model so as to further reduce modeling complexity. In addition, we are interested in exploring the new technology in deep learning (e.g. attention model) into feature extraction and correlation analysis of visual and textual features.

REFERENCES

[1] Y. Verma and C. V. Jawahar, "Image annotation by propagating labels from semantic neighbourhoods," *Int. J. Comput. Vis.*, vol. 121, no. 1, pp. 126–148, Jan. 2017.

[2] P. K. Bhagat and P. Choudhary, "Image annotation: Then and now," *Image Vis. Comput.*, vol. 80, pp. 1–23, Dec. 2018.

[3] Y. Ma, Y. Liu, Q. Xie, and L. Li, "CNN-feature based automatic image annotation method," *Multimedia Tools Appl.*, vol. 78, no. 3, pp. 3767–3780, Feb. 2019.

[4] S. L. Feng, R. Manmatha, and V. Lavrenko, "Multiple Bernoulli relevance models for image and video annotation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2004, pp. 1002–1009.

[5] A. Makadia, V. Pavlovic, and S. Kumar, "Baselines for image annotation," *Int. J. Comput. Vis.*, vol. 90, no. 1, pp. 88–105, 2010.

[6] Z. Niu, G. Hua, Q. Tian, and X. Gao, "Visual topic network: Building better image representations for images in social media," *Comput. Vis. Image Understand.*, vol. 136, pp. 3–13, Jul. 2015.

[7] F. Monay and D. Gatica-Perez, "PLSA-based image auto-annotation: Constraining the latent space," in *Proc. 12th Annu. ACM Int. Conf. Multimedia*, H. Schulzrinne, N. Dimitrova, M. A. Sasse, S. B. Moon, and R. Lienhart, Eds. New York, NY, USA: ACM, 2004, pp. 348–351.

[8] Q. Cheng, Q. Zhang, P. Fu, C. Tu, and S. Li, "A survey and analysis on automatic image annotation," *Pattern Recognit.*, vol. 79, pp. 242–259, Jul. 2018.

[9] L.-J. Li and L. Fei-Fei, "What, where and who? Classifying events by scene and object recognition," in *Proc. IEEE 11th Int. Conf. Comput. Vis.*, Oct. 2007, pp. 1–8.

[10] L.-J. Li, R. Socher, and L. Fei-Fei, "Towards total scene understanding: Classification, annotation and segmentation in an automatic framework," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 2036–2043.

[11] D. Tian and Z. Shi, "A two-stage hybrid probabilistic topic model for refining image annotation," *Int. J. Mach. Learn. Cybern.*, vol. 11, no. 2, pp. 417–431, Feb. 2020.

[12] P. Duygulu, K. Barnard, J. F. G. de Freitas, and D. A. Forsyth, "Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary," in *Proc. 7th Eur. Conf. Comput. Vis.* (Lecture Notes in Computer Science), vol. 2353, A. Heyden, G. Sparr, M. Nielsen, and P. Johansen, Eds. Copenhagen, Denmark: Springer, 2002, pp. 97–112, doi: 10.1007/3-540-47979-1_7.

[13] S. N. M. Foumani and A. Nickabadi, "A probabilistic topic model using deep visual word representation for simultaneous image classification and annotation," *J. Vis. Commun. Image Represent.*, vol. 59, pp. 195–203, Feb. 2019.

[14] Y. Zheng, T. Takiguchi, and Y. Ariki, "Image annotation with concept level feature using PLSA+CCA," in *Proc. 17th Int. Conf. Multimedia Modeling* (Lecture Notes in Computer Science), vol. 6524, K. Lee, W. Tsai, H. M. Liao, T. Chen, J. Hsieh, and C. Tseng, Eds. Taipei, Taiwan: Springer, 2011, pp. 454–464, doi: 10.1007/978-3-642-17829-0_43.

[15] L. Song, M. Luo, J. Liu, L. Zhang, B. Qian, M. H. Li, and Q. Zheng, "Sparse multi-modal topical coding for image annotation," *Neurocomputing*, vol. 214, pp. 162–174, Nov. 2016.

[16] T. Uricchio, L. Ballan, L. Seidenari, and A. Del Bimbo, "Automatic image annotation via label transfer in the semantic space," *Pattern Recognit.*, vol. 71, pp. 144–157, Nov. 2017.

[17] R. Lienhart, S. Romberg, and E. Hörster, "Multilayer pLSA for multimodal image retrieval," in *Proc. 8th ACM Int. Conf. Image Video Retr.*, S. Marchand-Maillet and Y. Kompatsiaris, Eds. Santorini Island, Greece: ACM, 2009.

[18] R. Zhang, L. Zhang, X. Wang, and L. Guan, "Multi-feature pLSA for combining visual features in image annotation," in *Proc. 19th ACM Int. Conf. Multimedia*, K. S. Candan, S. Panchanathan, B. Prabhakaran, H. Sundaram, W. Feng, and N. Sebe, Eds. Scottsdale, AZ, USA: ACM, 2011, pp. 1513–1516.

[19] D. Putthividhy, H. T. Attias, and S. S. Nagarajan, "Topic regression multimodal latent Dirichlet allocation for image annotation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, San Francisco, CA, USA, Jun. 2010, pp. 3408–3415.

[20] S. Romberg, R. Lienhart, and E. Hörster, "Multimodal image retrieval-fusing modalities with multilayer multimodal pLSA," *Int. J. Multimedia Inf. Retr.*, vol. 1, no. 1, pp. 31–44, 2012.

[21] M. Koskela and J. Laaksonen, "Convolutional network features for scene recognition," in *Proc. 22nd ACM Int. Conf. Multimedia*, K. A. Hua, Y. Rui, R. Steinmetz, A. Hanjalic, A. Natsev, and W. Zhu, Eds. Orlando, FL, USA: ACM, 2014, pp. 1169–1172.

[22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR)*, Y. Bengio and Y. LeCun, Eds.San Diego, CA, USA: arXiv, 2015. [Online]. Available: http://arxiv.org/abs/1409.1556

[23] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "DeCAF: A deep convolutional activation feature for generic visual recognition," in *Proc. Int. Conf. Mach. Learn. (ICML)*, vol. 32, Jun. 2014, pp. 647–655.

[24] Y. Gong, Y. Jia, T. Leung, A. Toshev, and S. Ioffe, "Deep convolutional ranking for multilabel image annotation," in *2nd Int. Conf. Learn. Represent. (ICLR)*, Y. Bengio and Y. LeCun, Eds. Banff, AB, Canada: arXiv, 2014. [Online]. Available: http://arxiv.org/abs/1312.4894

[25] V. N. Murthy, S. Maji, and R. Manmatha, "Automatic image annotation using deep learning representations," in *Proc. 5th ACM Int. Conf. Multimedia Retr.*, A. G. Hauptmann, C. Ngo, X. Xue, Y. Jiang, C. Snoek, and N. Vasconcelos, Eds. Shanghai, China: ACM, 2015, pp. 603–606.

[26] J. Johnson, L. Ballan, and F. Li, "Love thy neighbors: Image annotation by exploiting image metadata," in *Proc. IEEE Int. Conf. Comput. Vis.*, Santiago, Chile, Dec. 2015, pp. 4624–4632.

[27] M. Zang, D. Wen, K. Wang, T. Liu, and W. Song, "A novel topic feature for image scene classification," *Neurocomputing*, vol. 148, pp. 467–476, Jan. 2015.

[28] B. Wu, W. Chen, P. Sun, W. Liu, B. Ghanem, and S. Lyu, "Tagging like humans: Diverse and distinct image annotation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7967–7975. http://openaccess.thecvf.com/content_cvpr_2018/html/Wu_Tagging_Like_Humans_CVPR_2018_paper.html

[29] Y. Niu, Z. Lu, J. Wen, T. Xiang, and S. Chang, "Multi-modal multi-scale deep learning for large-scale image annotation," *IEEE Trans. Image Process.*, vol. 28, no. 4, pp. 1720–1731, Apr. 2019.

[30] Z. Lu, Z. Fu, T. Xiang, P. Han, L. Wang, and X. Gao, "Learning from weak and noisy labels for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 3, pp. 486–500, Mar. 2017.

[31] J. Tang, "Automatic image annotation and object detection," Ph.D. dissertation, School Electron. Comp. Sci., Univ. Southampton, Southampton, U.K., 2008. [Online]. Available: http://eprints.soton.ac.uk/265835/
[32] Y. Verma and C. V. Jawahar, "Image annotation using metric learning in semantic neighbourhoods," in *Proc. 12th Eur. Conf. Comput. Vis.* (Lecture Notes in Computer Science), vol. 7574, A. W. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds. Florence, Italy: Springer, 2012, pp. 836–849, doi: 10.1007/978-3-642-33712-3_60.

HAIYU SONG received the B.S., M.S., and Ph.D. degrees in computer software and theory from Jilin University, in 1996, 2003, and 2012, respectively. He is currently an Associate Professor with the School of Computer Science and Engineering, Dalian Nationalities University, China. His current research interests include image understanding, computer vision, and machine learning.

PENGJIE WANG received the B.S. and M.S. degrees in computer software and theory from Jilin University, in 2001 and 2004, respectively, and the Ph.D. degree in computer application from Zhejiang University, in 2015. He is currently a Professor with the School of Computer Science and Engineering, Dalian Nationalities University, China. His current research interests include computer vision, virtual reality, and machine learning.

JIAN YUN received the B.S. and M.S. degrees in computer application from Inner Mongolia University, in 1997 and 2003, respectively, and the Ph.D. degree from Shanghai Normal University, in 2010. He is currently a Professor with the School of Computer Science and Engineering, Dalian Nationalities University, China. His current research interests include intelligent information processing and artificial intelligence.

WEI LI received the B.S. and M.S. degrees in computer software and theory from Jilin University, in 2002 and 2005, respectively. She is currently a Lecturer with the School of Computer Science and Engineering, Dalian Nationalities University, China. Her current research interests include image understanding, computer vision, and machine learning.

BO XUE is currently pursuing the B.S. degree with the School of Computer Science and Engineering, Dalian Nationalities University, China. His current research interests include computer vision, image processing, and deep learning.

GANG WU is currently pursuing the B.S. degree with the School of Computer Science and Engineering, Dalian Nationalities University, China. His current research interests include computer vision, image processing, and deep learning.