Multi-Modal Multi-Scale Deep Learning for Large-Scale Image Annotation

Yulei Niu, Zhiwu Lu, Ji-Rong Wen, Tao Xiang, and Shih-Fu Chang

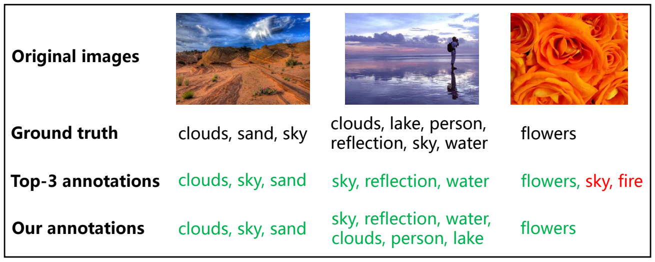
***Abstract*—Large-scale image annotation is a challenging task in image content analysis, which aims to annotate each image of a very large dataset with multiple class labels. In this paper, we focus on two main issues in large-scale image annotation: 1) how to learn stronger features for multifarious images; 2) how to annotate an image with an automatically-determined number of class labels. To address the first issue, we propose a multi-modal multi-scale deep learning model for extracting descriptive features from multifarious images. Specifically, the visual features extracted by a multi-scale deep learning subnetwork are refined with the textual features extracted from social tags along with images by a simple multi-layer perception subnetwork. Since we have extracted very powerful features by multi-modal multi-scale deep learning, we simplify the second issue and decompose large-scale image annotation into multi-class classification and label quantity prediction. Note that the label quantity prediction subproblem can be implicitly solved when a recurrent neural network (RNN) model is used for image annotation. However, in this paper, we choose to explicitly solve this subproblem directly using our deep learning model, resulting in that we can pay more attention to deep feature learning. Experimental results demonstrate the superior performance of our model as compared to the state-of-the-art (including RNN-based models).**

***Index Terms*—Large-scale image annotation, multi-scale deep learning, multi-modal deep learning, label quantity prediction.**

I. Introduction

IMage recognition [1]–[4] is a fundamental problem in image content analysis, which aims to assign class labels to images. Recent deep learning methods [5]–[9] have yielded exciting results on large-scale image recognition. The key point is that deep neural networks can encode the origin images to powerful visual features.

As a closely related task, image annotation [10]–[14] aims to annotate an image with multiple class labels, which can be solved using the multi-label classification methods [15], [16]. The main differences between image annotation and classic single-label image recognition lie in the class types and the label quantities. Firstly, the semantic classes in image annotation include not only objects, but also actions, activities and views (e.g. dancing, wedding and sunset). The features extracted to recognize objects may not be applied to scene recognition. Secondly, most of images consist of two or more semantic classes in the task of image annotation, and the quantity of class labels varies from one image to another. In this case, the traditional top-*k* models [10]–[14] may omit right labels or include wrong labels, as shown in Fig. 1.



Original images			
Ground truth	clouds, sand, sky	clouds, lake, person, reflection, sky, water	flowers
Top-3 annotations	clouds, sky, sand	sky, reflection, water	flowers, sky, fire
Our annotations	clouds, sky, sand	sky, reflection, water, clouds, person, lake	flowers

Fig. 1. Illustration of the difference between the traditional image annotation models that generate top-k annotations (e.g. $k = 3$) and our model that generates an automatically-determined number of annotations.

One trend in image annotation is that the image datasets become larger and larger [17]–[20]. The large-scale datasets add challenges into image annotation in two aspects: 1) the images are so multifarious that it is difficult to extract strong features for image annotation; 2) the quantity of class labels significantly varies from one image to another, which makes the traditional top-*k* models fail in this case. In other words, there exist two main issues in large-scale image annotation: 1) how to learn stronger features for multifarious images; 2) how to annotate an image with an automatically-determined number of class labels (see Fig. 1). In this paper, we focus on addressing the two main issues to improve the traditional large-scale image annotation methods [17]–[20].

Due to the latest advances in deep learning [21], [22], many deep models have been proposed for large-scale image annotation [23]–[27]. The main contributions of these recent works can be summarized as follows. Firstly, strong features are the most fundamental factor for image annotation. Most of recent works explored convolution neural network (CNN) [5]–[9] in extracting stronger visual features. For instance, Johnson *et al.* [23] combined the visual features extracted by CNN from one image with those from other similar images. Secondly, side information can be used to improve the performance of image annotation. Users often loosely annotate social images, and side information (i.e. tags) can be recorded into image metadata for image annotation [24]. In order to seek similar neighborhood images, Johnson *et al.* [23] utilized tags from image metadata as side information. Besides, Hu *et al.* [25] formed a multi-layer group-concept-tag graph using both tags and group labels. Last but not least, recurrent neural network

Y. Niu, Z. Lu, and J.-R. Wen are with the Beijing Key Laboratory of Big Data Management and Analysis Methods, School of Information, Renmin University of China, Beijing 100872, China (email: niu@ruc.edu.cn, luzhiwu@ruc.edu.cn, jrwen@ruc.edu.cn).

T. Xiang is with the School of Electronic Engineering and Computer Science, Queen Mary University of London, Mile End Road, London E1 4NS, United Kingdom (email: t.xiang@qmul.ac.uk).

S.-F. Chang is with the Department of Electrical Engineering, Columbia University, New York, NY 10027, USA. (email: sc250@columbia.edu).

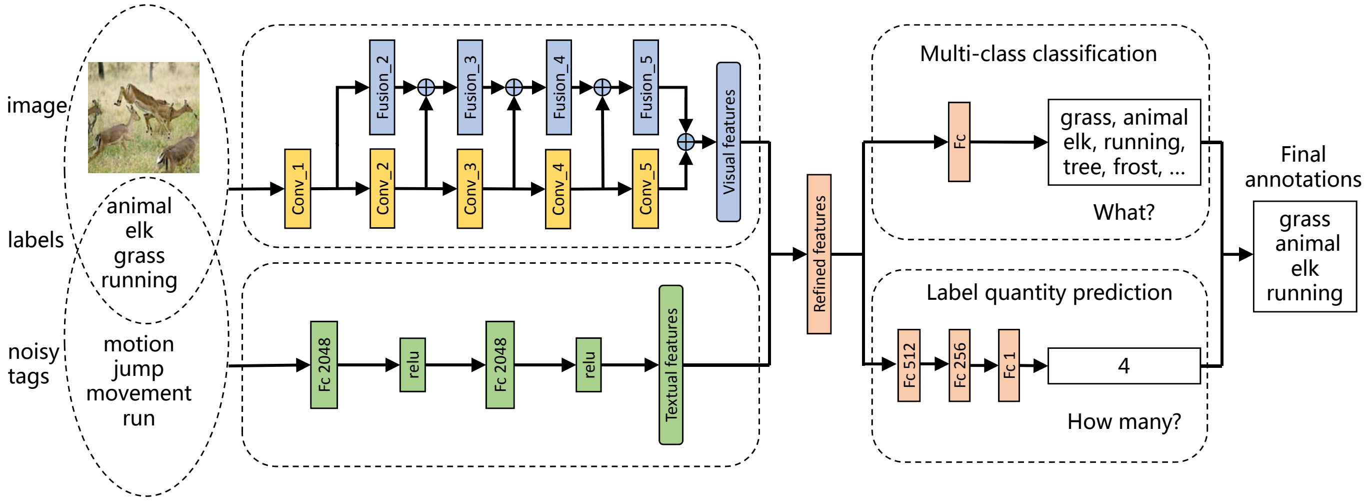


Fig. 2. The flowchart of our multi-modal multi-scale deep learning model for large-scale image annotation. In this model, four components are included: visual feature learning, textual feature learning, multi-class classification, and label quantity prediction.

(RNN) [28]–[30] has been combined with CNN for large-scale image annotation [24], [26], [27], where the label quantity prediction subproblem can be implicitly solved using RNN. Specifically, CNN encodes raw pixels of images into visual feature, while RNN decodes the visual feature into a sequential label prediction path for image annotation.

Inspired by the above closely related works, we propose a multi-modal multi-scale deep learning model to address the two main issues in large-scale image annotation, as shown in Fig. 2. Specifically, to learn stronger features from multifarious images, we combine the visual features extracted by a multi-scale deep learning subnetwork with the textual features extracted from social tags along with images by a simple multi-layer perception subnetwork. In this paper, following the ideas of [31]–[34], the multi-scale deep learning subnetwork is defined based on ResNet-101 [5] (see Fig. 2): the multi-scale features are first extracted from different levels of layers, and a fusion block is further developed to combine the multi-scale features with the original ones. In this way, the visual information can be transferred from low-level features to high-level features iteratively. Since we have extracted very powerful features by multi-modal multi-scale deep learning, we simplify the second issue and decompose large-scale image annotation into multi-class classification and label quantity prediction. Although the label quantity prediction subproblem can be implicitly solved by adopting RNN for image annotation, we choose to explicitly solve this subproblem directly using our deep learning model (see Fig. 2). This enables us to pay more attention to deep feature learning. In the end, the results of multi-class classification and predicted label quantity are combined to determine the final annotations. Note that our model is different form the traditional top-*k* models [10]–[14] that our model can automatically determine the number of class labels for each image (see Fig. 1).

The flowchart of our multi-modal multi-scale deep learning model is shown in Fig. 2, where four components are included: visual feature learning, textual feature learning, multi-class classification, and label quantity prediction. To evaluate the performance of our model, we conduct extensive experiments on two benchmark datasets: NUS-WIDE [35] and MSCOCO [36]. The experimental results demonstrate the superior performance of our model as compared to the state-of-the-art models (including CNN-RNN [24], [26], [27]). In addition, the main components of our model (see Fig. 2) are also shown to be effective in large-scale image annotation.

our main contributions can be summarized as follows:

- We have proposed a multi-scale ResNet model for visual feature learning, which is shown to outperform the original ResNet model in large-scale image annotation.
- We have effectively exploited the social tags for large-scale image annotation by defining a simple multi-layer perception model for textual feature learning.
- We have explicitly solved the label quantity prediction subproblem using the features exacted by our model, which has been rarely considered. This enables us to pay more attention to deep feature learning.

The remainder of this paper is organized as follows. Section II provides related works on large-scale image annotation. Section III gives the details of the proposed model for large-scale image annotation. Experimental results are presented in Section IV. Finally, the conclusions are drawn in Section V.

II. Related Work

A. Multi-Scale CNN Models

Feature extraction is crucial for different applications in image content analysis. A remarkable model is expected to encode the raw pixels of images into powerful visual features. In the past few years, CNN models have been widely applied to single-label image classification [5]–[9] due to their outstanding performance. However, only the highest-level features with global information are used for image classification, without considering other low-level features with local information. In other words, multi-scale features are not employed. Recent researches begin to focus on how to make full use of different levels of features in different applications. Gong *et al.* [31] noticed that the robustness of global features was limited due to the lack of geometric invariance, and thus proposed a multi-scale orderless pooling (MOP-CNN) scheme, which concentrates orderless Vectors of Locally Aggregated Descriptors (VLAD) pooling of CNN activations at multiple levels. Yang and Ramanan [32] argued that different scales of features can be used in different image classification tasks through multi-task learning, and developed directed acyclic graph structured CNNs (DAG-CNNs) to extract multi-scale features for both coarse and fine-grained classification tasks. Cai *et al.* [33] presented a multi-scale CNN (MS-CNN) model for fast object detection, which performs object detection using both lower and higher output layers. Liu *et al.* [34] proposed a multi-scale triplet CNN model for person re-identification. The results reported in [31]–[34] have shown that multi-scale features are indeed effective for image content analysis. In this paper, we follow the ideas of [31]–[34], but adopt a different method for multi-scale feature learning.

B. Image Annotation Using Side Information

In addition to pixels of images, social information provided by social users has also been utilized to improve the performance of large-scale image annotation, including noisy tags [23]–[25] and group labels [25], [37], [38]. Johnson *et al.* [23] found neighborhoods of images nonparametrically according to the image metadata, and combined the visual features of each image and its neighborhoods. Liu *et al.* [24] filtered noisy tags to obtain true tags, which serve as the supervision to train the CNN model and also set the initial state of RNN. Hu *et al.* [25] utilized both tags and group labels to form a multi-layer group-concept-tag graph, which can encode the diverse label relations. Different from [25] that adopted a deep model, the group labels were used to learn the context information for image annotation in [37], [38], but without considering deep models. In this paper, we will make attempt to extract powerful textual features from noisy tags to boost the visual features extracted by CNN models.

C. Label Quantity Prediction for Image Annotation

Extended from single-label image classification task, multi-label image annotation aims to recognize one or more potential classes in each image. Early works [23], [25], [39] focused on classification, and assigned top-k class labels to each image for annotation. However, the quantities of class labels in different images vary significantly, and the top-k annotations degrade the performance of image annotation. To overcome this issue, recent works start to predict the quantity of class labels. The most popular method is the CNN-RNN architecture, where the CNN subnetwork encodes the input pixels of images into visual feature, and the RNN subnetwork decodes the visual feature into a label prediction path [24], [26], [27]. Specifically, RNN can not only perform classification, but also predict the quantity of class labels. Since RNN requires an ordered sequential list as input, the unordered label set is expected to be transformed in advance based on the rare-first rule [27] or the frequent-first rule [26], which means that we need to obtain priori knowledge from the training set.

III. The Proposed Model

Our multi-modal multi-scale deep learning model for large-scale image annotation can be divided into four components: visual feature learning, textual feature learning, multi-class classification, and label quantity prediction, as shown in Fig. 2. Specifically, a multi-scale CNN subnetwork is proposed to extract visual feature from raw image pixels, and a multi-layer perception subnetwork is applied to extract textual features from noisy tags. The joint visual and textual features are connected to a simple fully connected layer for multi-class classification. To annotate each image more precisely, we utilize another multi-layer perception subnetwork to predict the quantity of class labels. The results of multi-class classification and label quantity prediction are finally merged for image annotation. In the following, we will first give the details of the four components of our model, and then provide the algorithms for model training and test process.

A. Multi-Scale CNN for Visual Feature Learning

Our multi-scale CNN subnetwork consists of two parts: the basic CNN model which encodes an image to both low-level and high-level features, and the feature fusion model which combines multi-scale features iteratively.

Basic CNN Model. Given the raw pixels of an image $\mathbf{I}$, the basic CNN model encodes them into K levels of feature maps $\mathbf{M}_1, \mathbf{M}_2, \cdots, \mathbf{M}_K$ through a series of scales $S_1(\cdot), S_2(\cdot), \cdots, S_K(\cdot)$. Each scale can be a composite function of operations, such as convolution (Conv), pooling (Pooling), batch normalization (BN), and activation function. The encoding process can be formulated as:

$$\begin{aligned} &\mathbf{M}_0 = \mathbf{I} \\ &\mathbf{M}_l = S_l(\mathbf{M}_{l-1}), l = 1, 2, ..., K \end{aligned} \tag{1}$$

For the basic CNN model, the last feature map $\mathbf{M}_K$ is often used to produce the final feature vector, e.g., the conv5_3 layer of the ResNet-101 model [5].

Feature Fusion Model. When multi-scale feature maps $\mathbf{M}_1, \mathbf{M}_2, \cdots, \mathbf{M}_K$ are obtained, the feature fusion model is proposed to combine these original feature maps iteratively via a set of fusion functions $\{F_l(\cdot)\}$, as shown in Fig. 3. The fused feature map $\widetilde{\mathbf{M}}_l$ is formulated as:

$$\begin{aligned} &\widetilde{\mathbf{M}}_1 = \mathbf{M}_1 \\ &\widetilde{\mathbf{M}}_l = \mathbf{M}_l + F_l(\widetilde{\mathbf{M}}_{l-1}), l = 2, ..., K \end{aligned} \tag{2}$$

In this paper, we define the fusion function $F_l(\cdot)$ as:

$$F_l(\cdot) = \phi_{l2}\left(\phi_{l1}\left(\cdot\right)\right) \tag{3}$$

where $\phi_{l1}(\cdot)$ and $\phi_{l2}(\cdot)$ are two composite functions consisting of three consecutive operations: a convolution (Conv), followed by a batch normalization (BN) and a rectified linear unit (ReLU). The difference between $\phi_{l1}(\cdot)$ and $\phi_{l2}(\cdot)$ lies in the convolution layer. The 3×3 Conv in $\phi_{l1}(\cdot)$ is used to guarantee that $\mathbf{M}_l$ and $F_l(\widetilde{\mathbf{M}}_{l-1})$ have the same height and weight, while the 1×1 Conv in $\phi_{l2}(\cdot)$ can not only increase dimensions and interact information between different channels, but also reduce the number of parameters and improve computational

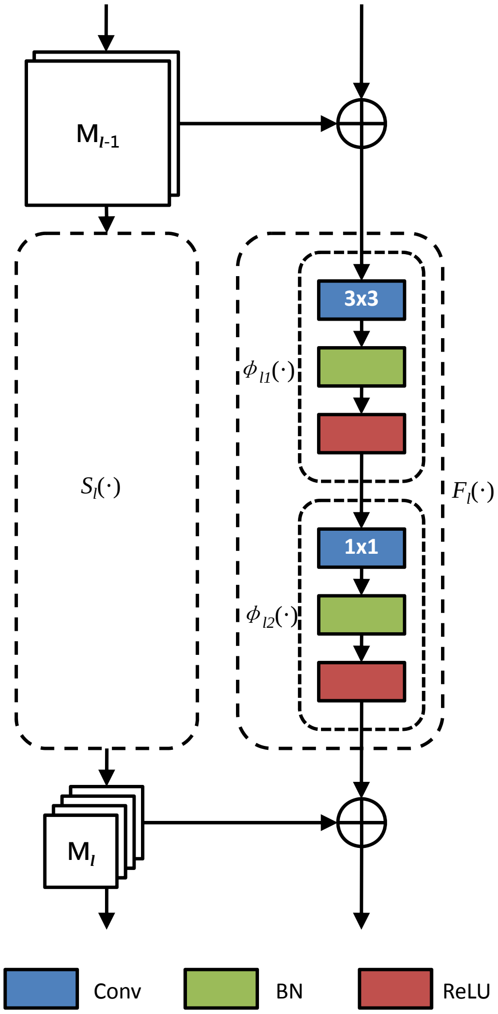


Fig. 3. The block architecture of the feature fusion model.

efficiency [5], [8]. At the end of the fused feature map $\widetilde{\mathbf{M}}_K$, an average pooling layer is used to extract the final visual feature vector $\mathbf{f}_v$ for image annotation.

In this paper, we take ResNet-101 [5] as the basic CNN model, and select the final layers of five scales (*i.e.*, conv1, conv2_3, conv3_4, conv4_23 and conv5_3, totally 5 final convolutional layers) for multi-scale fusion. In particular, the feature maps at the five scales have the size of 112×112, 56×56, 28×28, 14×14 and 7×7, along with 64, 256, 512, 1024 and 2048 channels, respectively. The architecture of our multi-scale CNN subnetwork is shown in Fig. 4.

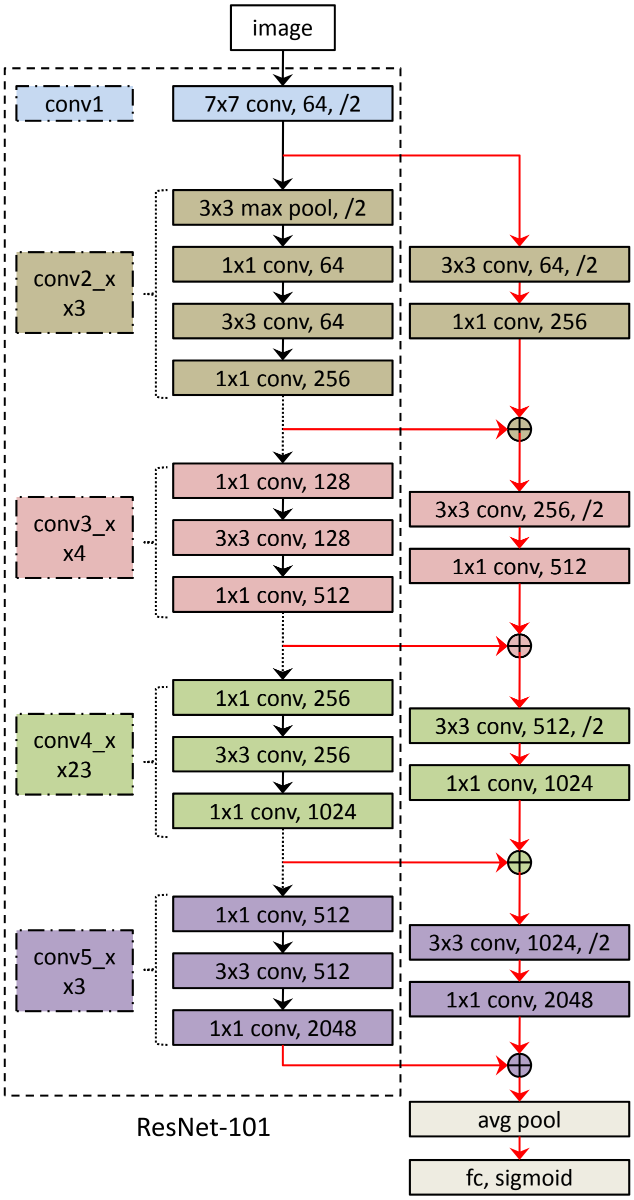


Fig. 4. The architecture of our multi-scale CNN subnetwork. Here, ResNet-101 is used as the basic CNN model, and four feature fusion blocks are included for multi-scale feature learning.

B. Multi-Layer Perception Model for Textual Feature Learning

We further investigate how to learn textual features from noisy tags provided by social users. The noisy tags of image $\mathbf{I}$ are represented as a one-hot vector $\mathbf{t} = (t_1, t_2, \cdots, t_T)$, where T is the volume of tags, and $t_j = 1$ if image $\mathbf{I}$ is annotated with tag j. Since the one-hot vector is sparse and noisy, we encode the raw vector into a dense textual feature vector $\mathbf{f}_t$ using a multi-layer perception model, which consists of two hidden layers (each with 2,048 units), as shown in Fig. 2. Note that only a simple neural network model is used for textual feature learning. Our consideration is that the noisy tags have a high-level semantic information and a complicated model would degrade the performance of textual feature learning. This observation has also been reported in [40]–[42] in the filed of natural language processing.

In this paper, the visual feature vector $\mathbf{f}_v$ and the textual feature vector $\mathbf{f}_t$ are forced to have the same dimension, which enables them to play the same important role in feature learning. By taking a multi-modal feature learning strategy, we concatenate the visual and textual feature vectors $\mathbf{f}_v$ and $\mathbf{f}_t$ into a joint feature vector $\mathbf{f} = [\mathbf{f}_v, \mathbf{f}_t]$ for the subsequent multi-class classification and label quantity prediction.

```
Algorithm 1: Model Training
Input: The training set, the ground-truth class labels of
       training images, the noisy tags of training images
Output: Model parameters
Finetune the basic CNN model using the training images
  and their ground-truth class labels;
Initialize the parameters of the multi-scale CNN
  (MS-CNN) model with the finetuned CNN;
Train MS-CNN using raw pixels and ground-truth class
  labels for visual feature learning;
Train the multi-layer perception (MLP) model using
  noisy tags and ground-truth class labels for textual
  feature learning;
Fix the parameters of well-trained MS-CNN and MLP,
  train the multi-class classification and label quantity
  prediction models for image annotation;
Return all parameters.
```

C. Multi-Class Classification and Label Quantity Prediction

When we have extracted the joint visual and textual feature vector from raw pixels and noisy tags, we introduce the details of image annotation in the following, which includes multi-class classification and label quantity prediction.

Multi-Class Classification. Since each image can be annotated with one or more categories, we define a multi-class classifier for image annotation. Specifically, the joint visual and textual feature is connected to a fully connected layer for logit calculation, followed by a sigmoid function for probability calculation, as shown in Fig. 2.

Label Quantity Prediction. To improve the performance of image annotation and also evaluate the effectiveness of our multi-modal multi-scale deep learning under fair settings, we make attempt to predict the quantity of class labels. Specifically, we formulate label quantity prediction as a regression problem, and adopt a multi-layer perception model as the regressor. As shown in Fig. 2, the regressor consists of two hidden layers with 512 and 256 units, respectively. To avoid the overfitting problem, the dropout layers are applied in all hidden layers with the dropout rate 0.5.

Note that we have explicitly solved the label quantity prediction subproblem using the features exacted by our multi-modal multi-scale deep learning model. Although label quantity prediction in an explicit manner has been rarely considered in the literature, its effectiveness in image annotation has been verified by our experimental results (see Tables I and II). The greatest advantage of label quantity prediction is that we can pay more attention to deep feature learning.

D. Model Training

During the process of model training, we apply a multi-stage strategy and divide the architecture into several branches: visual features learning, textual features learning, multi-class classification, and label quantity prediction. Specifically, the original ResNet model is fine-tuned with a small learning rate. When the fine-tuning is finished, we will fix the parameters of ResNet and train the multi-scale blocks. In this paper, the training of the textual model is separated from the visual model, and thus the two models can be trained synchronously. After visual and textual feature learning, we fix the parameters of the visual and textual models, and train the multi-class classification and label quantity prediction models separately. The complete training process is outlined in Algorithm 1.

```
Algorithm 2: Test Process
Input: Test image $\mathbf{I}$, the noisy tags of $\mathbf{I}$
Output: Predicted class labels
Extract the visual feature vector $\mathbf{f}_v$ from test image $\mathbf{I}$
  with the MS-CNN model;
Extract the textual feature vector $\mathbf{f}_t$ from the noisy tags
  of $\mathbf{I}$ with the MLP model;
Concatenate $\mathbf{f}_v$ and $\mathbf{f}_t$ to a joint feature vector $\mathbf{f}$;
Predict the probabilities of labels $\mathbf{p}$;
Predict the label quantity $\hat{m}$;
Select top $\hat{m}$ class labels according to $\mathbf{p}$;
Return the selected $\hat{m}$ class labels.
```

To provide further insights on model training, we define the loss functions for training the four branches as follows.

Sigmoid Cross Entropy Loss. For training the first three branches, the features are first connected to a fully connected layer to compute the logits $\hat{\mathbf{l}}$, and a sigmoid cross entropy loss is then applied for model training:

$$L_1 = \sum \mathbf{y} \cdot \Big(-\log(\sigma(\hat{\mathbf{l}}))\Big) + (1-\mathbf{y}) \cdot \Big(-\log(1-\sigma(\hat{\mathbf{l}}))\Big) \quad (4)$$

where $\mathbf{y}$ is the one-hot vector that collects the ground-truth labels of image $\mathbf{I}$ and $\sigma(\cdot)$ is the sigmoid function.

Mean Squared Error Loss. For training the label quantity prediction model, the features are also connected to a fully connected layer with one output unit to compute the predicted label quantity $\hat{m}$. We then apply the following mean squared error loss for model training:

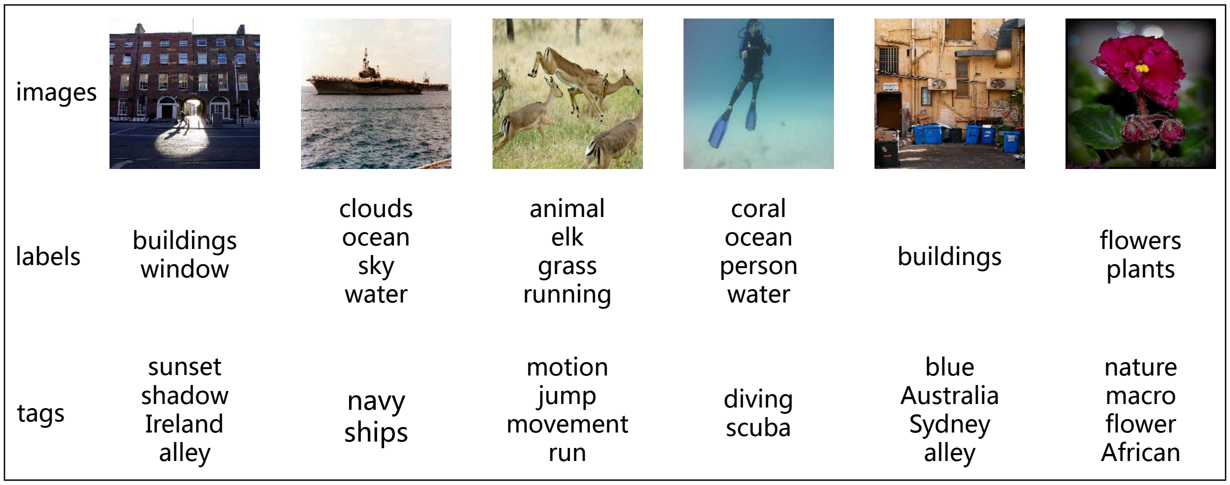
$$L_2 = \sum (\hat{m} - m)^2 \quad (5)$$

where m is the ground-truth label quantity of image $\mathbf{I}$.

In this paper, the four branches of our model are not trained in a single process. Our main consideration is that our model has a too complicated architecture and thus the overfitting issue still exists even if a large dataset is provided for model training. In the future work, we will make efforts to develop a robust algorithm for train our model in a single process.

E. Test Process

At the test time, we first extract the joint visual and textual feature vector from each test image, and then execute multi-class classification and label quantity prediction synchronously. When the predicted probabilities of labels $\mathbf{p}$ and the predicted label quantity $\hat{m}$ have been obtained for the test image, we select the top $\hat{m}$ candidates as our final annotations. The complete test process is outlined in Algorithm 2.



images						
labels	buildings window	clouds ocean sky water	animal elk grass running	coral ocean person water	buildings	flowers plants
tags	sunset shadow Ireland alley	navy ships	motion jump movement run	diving scuba	blue Australia Sydney alley	nature macro flower African

Fig. 5. Examples of the NUS-WIDE dataset. Images (first row) are followed with class labels (second row) and noisy tags (third row).

IV. Experiments

A. Datasets and Settings

1) Datasets: The two most widely used benchmark datasets for large-scale image annotation are selected for performance evaluation. The first dataset is NUS-WIDE [35], which consists of 269,648 images, 81 class labels, and 1,000 most frequent noisy tags from Flickr image metadata. The number of Flickr images varies in different studies, since some Flickr links are going invalid. For fair comparison, we also remove images without any social tag. As a result, we obtain 94,570 training images and 55,335 test images. Some examples of this dataset are shown in Fig. 5. The second dataset is MSCOCO [36], which consists of 87,188 images, 80 class labels, and 1,000 most frequent noisy tags from the 2015 MSCOCO image captioning challenge. The training/test split of the MSCOCO dataset is 56,414/30,774.

2) Evaluation Metrics: The per-class and per-image metrics including precision and recall have been widely used in related works [23]–[27], [39]. In this paper, the per-class precision (C-P) and per-class recall (C-R) are obtained by computing the mean precision and recall over all the classes, while the overall per-image precision (I-P) and overall per-image recall (I-R) are computed by averaging over all the test images. Moreover, the per-class F1-score (C-F1) and overall per-image F1-score (I-F1) are used for comprehensive performance evaluation by combining precision and recall with the harmonic mean. The six metrics are defined as follows:

$$\begin{aligned} \text{C-P} &= \frac{1}{C}\sum_{j=1}^{C}\frac{\text{NI}_j^c}{\text{NI}_j^p}, \quad \text{I-P} = \frac{\sum_{i=1}^{N}\text{NL}_i^c}{\sum_{i=1}^{N}\text{NL}_i^p} \\ \text{C-R} &= \frac{1}{C}\sum_{j=1}^{C}\frac{\text{NI}_j^c}{\text{NI}_j^g}, \quad \text{I-R} = \frac{\sum_{i=1}^{N}\text{NL}_i^c}{\sum_{i=1}^{N}\text{NL}_i^g} \\ \text{C-F1} &= \frac{2\cdot\text{C-P}\cdot\text{C-R}}{\text{C-P}+\text{C-R}}, \quad \text{I-F1} = \frac{2\cdot\text{I-P}\cdot\text{I-R}}{\text{I-P}+\text{I-R}} \end{aligned} \tag{6}$$

where C is the number of classes, N is the number of test images, NI_j^c is the number of images correctly labelled as class j, NI_j^g is the number of images that have a ground-truth label of class j, NI_j^p is the number of images predicted as class j, NL_i^c is the number of correctly annotated labels for image i, NL_i^g is the number of ground-truth labels for image i, and NL_i^p is the number of predicted labels for image i.

According to [39], the above per-class metrics are biased toward the infrequent classes, while the above per-image metrics are biased toward the frequent classes. Similar observations have also been presented in [43]. As a remedy, following the idea of [43], we define a new metric called H-F1 with the harmonic mean of C-F1 and I-F1:

$$\text{H-F1} = \frac{2\cdot\text{C-F1}\cdot\text{I-F1}}{\text{C-F1}+\text{I-F1}} \tag{7}$$

Since H-F1 takes both per-class and per-image F1-scores into account, it enables us to make easy comparison between different methods for large-scale image annotation.

3) Settings: In this paper, the basic CNN module makes use of ResNet-101 [5] which is pretrained on the ImageNet 2012 classification challenge dataset [44]. Our experiments are all conducted on TensorFlow. The input images are resized to 224×224 pixels. For training the basic CNN model, the multi-scale CNN model, and the multi-class classifier, the learning rate is set to 0.001 for NUS-WIDE and 0.002 for MSCOCO, respectively. For training the textual feature learning model and the label quantity prediction model, the learning rate is set to 0.1 for NUS-WIDE and 0.0001 for for MSCOCO, respectively. These models are trained using a Momentum with the momentum rate of 0.9. The decay rate of batch normalization and weight decay is set to 0.9997.

4) Compared Methods: We conduct two groups of experiments for evaluation and choose competitors to compare accordingly: (1) We first make comparison among several variants of our complete model shown in Fig. 2 by removing one or more components from our model, so that the effectiveness of each component can be evaluated properly. (2) To compare with a wider range of image annotation methods, we also compare with the published results on the two benchmark datasets. These include the state-of-the-art deep learning methods for large-scale image annotation.

TABLE I
EFFECTIVENESS EVALUATION FOR THE MAIN COMPONENTS OF OUR MODEL ON THE NUS-WIDE DATASET.

Models	Multi-Modal	Quality Prediction	C-P (%)	C-R (%)	C-F1 (%)	I-P (%)	I-R (%)	I-F1 (%)	H-F1 (%)
Upper bound (Ours)	yes	yes	81.46	75.83	78.55	85.87	85.87	85.87	82.05
MS-CNN+Tags+LQP	yes	yes	**80.27**	**60.95**	**69.29**	**84.55**	**76.80**	**80.49**	**74.47**
MS-CNN+LQP	no	yes	65.00	52.79	58.26	77.10	73.77	75.40	65.73
MS-CNN+Tags	yes	no	57.15	64.39	60.55	62.85	74.03	67.98	64.05
MS-CNN	no	no	49.60	56.36	52.77	60.07	70.77	64.98	58.24
CNN	no	no	45.86	57.34	50.96	59.88	70.54	64.77	57.04

TABLE II
EFFECTIVENESS EVALUATION FOR THE MAIN COMPONENTS OF OUR MODEL ON THE MSCOCO DATASET.

Models	Multi-Modal	Quality Prediction	C-P (%)	C-R (%)	C-F1 (%)	I-P (%)	I-R (%)	I-F1 (%)	H-F1 (%)
Upper bound (Ours)	yes	yes	75.43	71.23	73.27	77.15	77.15	77.15	75.16
MS-CNN+Tags+LQP	yes	yes	**74.88**	**64.96**	**69.57**	**76.34**	**70.25**	**73.17**	**71.32**
MS-CNN+LQP	no	yes	67.48	60.93	64.04	70.22	67.93	69.06	66.46
MS-CNN+Tags	yes	no	63.69	63.87	63.78	64.47	68.77	66.55	65.14
MS-CNN	no	no	58.47	58.77	58.62	61.06	62.15	63.02	60.74
CNN	no	no	57.16	57.24	57.20	60.23	64.25	62.17	59.58

B. Effectiveness Evaluation for Model Components

We conduct the first group of experiments to evaluate the effectiveness of the main components of our model for large-scale image annotation. Five closely related models are included in component effectiveness evaluation: (1) CNN denotes the original ResNet-101 model; (2) MS-CNN denotes the multi-scale ResNet model shown in Fig. 3; (3) MS-CNN+Tags denotes the multi-modal multi-scale ResNet model that learns both visual and textual features for image annotation; (4) MS-CNN+LQP denotes the multi-scale ResNet model that performs both multi-class classification and label quantity prediction (LQP) for image annotation; (5) MS-CNN+Tags+LQP denotes our complete model shown in Fig. 2. Note that the five models can be recognized by whether they are multi-scale, multi-modal, or making LQP. This enables us to evaluate the effectiveness of each component of our model.

The results on the two benchmark datasets are shown in Tables I and II, respectively. Here, we also show the upper bounds of our complete model (i.e. MS-CNN+Tags+LQP) obtained by directly using the ground-truth label quantities to refine the predicted annotations (without LQP). We can make the following observations: (1) Although label quantity prediction has been explicitly addressed in our model (unlike RNN), it indeed leads to significant improvements according to the H-F1 score (10.42 percent for NUS-WIDE, and 6.18 percent for MSCOCO), when MS-CNN+Tags is used for feature learning. The improvements achieved by label quantity prediction become smaller when only MS-CNN is used for feature learning. This is because that the quality of label quantity prediction would degrade when the social tags are not used as textual features. (2) The social tags are crucial not only for label quantity prediction but also for the final image annotation. Specifically, the textual features extracted from social tags yield significant gains according to the H-F1 score (8.74 percent for NUS-WIDE, and 4.86 percent for MSCOCO), when both MS-CNN and LQP are adopted for image annotation. This is also supported by the gains achieved by MS-CNN+Tags over MS-CNN. (3) The effectiveness of MS-CNN is verified by the comparison MS-CNN vs. CNN. Admittedly only small gains ($> 1\%$ in terms of H-F1) have been obtained by MS-CNN. However, this is still impressive since ResNet-101 is a very powerful CNN model. In summary, we have evaluated the effectiveness of all the components of our complete model (shown in Fig. 2).

C. Comparison to the State-of-the-Art Methods

In this group of experiments, we compare our method with the state-of-the-art deep learning methods for large-scale image annotation. The following competitors are included: (1) CNN+Logistic [25]: This model fits a logistic regression classifier for each class label. (2) CNN+Softmax [39]: It is a CNN model that uses softmax as classifier and the cross entropy as the loss function. (3) CNN+WARP [39]: It uses the same CNN model as above, but a weighted approximate ranking loss function is adopted for training to promote the prec@K metric. (4) CNN-RNN [27]: It is a CNN-RNN model that uses output fusion to merge CNN output features and RNN outputs. (5) RIA [26]: In this CNN-RNN model, the CNN output features are used to set the hidden state of Long short-term memory (LSTM) [45]. (6) TagNeighbour [23]: It uses a non-parametric approach to find image neighbours according to metadata, and then aggregates image features for classification. (7) SINN [25]: It uses different concept layers of tags, groups, and labels to model the semantic correlation between concepts of different abstraction levels, and a bidirectional RNN-like algorithm is developed to integrate information for prediction. (8) SR-NN-RNN [24]: This CNN-RNN model uses a semantically regularised embedding layer as the interface between the CNN and RNN.

The results on the two benchmark datasets are shown in Tables III and IV, respectively. Here, we also show the upper bounds of our model obtained by directly using the ground-truth label quantities to refine the predicted annotations (without LQP). It can be clearly seen that our model outperforms the state-of-the-art deep learning methods according to the H-F1 score. This provides further evidence that our multi-modal multi-scale CNN model along with label quantity prediction is indeed effective in large-scale image annotation. Moreover,

TABLE III
COMPARISON TO THE STATE-OF-THE-ART ON THE NUS-WIDE DATASET.

Methods	Multi-Modal	Quality Prediction	C-P (%)	C-R (%)	C-F1 (%)	I-P (%)	I-R (%)	I-F1 (%)	H-F1 (%)
Upper bound (Ours)	yes	yes	81.46	75.83	78.55	85.87	85.87	85.87	82.05
Ours	yes	yes	**80.27**	60.95	**69.29**	**84.55**	76.80	**80.49**	**74.47**
SR-CNN-RNN [24]	yes	yes	71.73	**61.73**	66.36	77.41	76.88	77.15	71.35
SINN [25]	yes	yes	58.30	60.30	59.44	57.05	**79.12**	66.30	62.68
TagNeighboor [23]	yes	no	54.74	57.30	55.99	53.46	75.10	62.46	59.05
RIA [26]	no	yes	52.92	43.62	47.82	68.98	66.75	67.85	56.10
CNN-RNN [27]	no	no	40.50	30.40	34.70	49.90	61.70	55.20	42.61
CNN+WARP [39]	no	no	31.65	35.60	33.51	48.59	60.49	53.89	41.32
CNN+softmax [39]	no	no	31.68	31.22	31.45	47.82	59.52	53.03	39.48
CNN+logistic [25]	no	no	45.60	45.03	45.31	51.32	70.77	59.50	51.44

TABLE IV
COMPARISON TO THE STATE-OF-THE-ART ON THE MSCOCO DATASET.

Methods	Multi-Modal	Quality Prediction	C-P (%)	C-R (%)	C-F1 (%)	I-P (%)	I-R (%)	I-F1 (%)	H-F1 (%)
Upper bound (Ours)	yes	yes	75.43	71.23	73.27	77.15	77.15	77.15	75.16
Ours	yes	yes	**74.88**	**64.96**	**69.57**	76.34	70.25	73.17	**71.32**
SR-CNN-RNN [24]	yes	yes	71.38	63.13	67.00	**77.41**	**73.05**	**75.16**	70.84
RIA [26]	no	yes	64.32	54.07	58.75	74.20	64.57	69.05	63.48
CNN-RNN [27]	no	no	66.00	55.60	60.40	69.20	66.40	67.80	63.89
CNN+WARP [39]	no	no	52.50	59.30	55.70	61.40	59.80	60.70	58.09
CNN+softmax [39]	no	no	57.00	59.00	58.00	62.10	60.20	61.10	59.51
CNN+logistic [25]	no	no	59.30	58.60	58.90	61.70	65.00	63.30	61.02

TABLE V
RESULTS OF LABEL QUANTITY PREDICTION (LQP) WITH DIFFERENT GROUPS OF FEATURES ON THE NUS-WIDE DATASET.

Features for LQP	Accuracy (%)	MSE
MS-CNN	45.92	0.673
MS-CNN+Tags	**48.26**	**0.437**

TABLE VI
RESULTS OF LABEL QUANTITY PREDICTION WITH DIFFERENT GROUPS OF FEATURES ON THE MSCOCO DATASET.

Features for LQP	Accuracy (%)	MSE
MS-CNN	46.27	0.773
MS-CNN+Tags	**47.96**	**0.564**

it is also noted that our model with explicit label quantity prediction yields better results than the CNN-RNN models with implicit label quantity prediction (i.e. SR-CNN-RNN, RIA, and SINN). This shows that RNN is not the unique model suitable for label quantity prediction. In particular, when it is done explicitly like our model, we can pay more attention to the CNN model itself for deep feature learning. Considering that RNN needs prior knowledge on class labels, our model is expected to have a wider use in real-word applications. In addition, the annotation methods that adopt multi-modal feature learning or label quantity prediction are generally shown to outperform the methods without considering any of the two components for image annotation.

D. Further Evaluation

1) Results of Label Quantity Prediction: We have evaluated the effectiveness of label quantity prediction in the above experiments, but have not shown the quality of label quantity prediction itself. In this paper, to measure the quality of label quantity prediction, two metrics are computed as follows: 1) Accuracy: the predicted label quantities are first quantized to their nearest integers, and then compared to the ground-truth label quantities to obtain the accuracy; 2) Mean Squared Error (MSE): the mean squared error is computed by directly comparing the predicted label quantities to the the ground-truth ones. The results of label quantity prediction on the two benchmark datasets are shown in Tables V and VI, respectively. It can be seen that more than 45% label quantities are correctly predicted in all cases. Moreover, the textual features extracted from social tags are shown to yield significant gains when MSE is used as the measure.

2) Qualitative Results of Image Annotation: Fig. 6 shows several annotation examples on NUS-WIDE when our model is employed. Here, annotations with green font are included in ground-truth class labels, while annotations with blue font are correctly tagged but not included in ground-truth class labels (i.e. the ground-truth labels may be incomplete). It is observed that the images in the first two rows are all annotated exactly correctly, while this is not true for the images in the last two rows. However, by checking the extra annotations (with blue font) generated for the the images in the last two rows, we find that they are all consistent with the content of the images.

3) Training and Test Time: We provide the training and test time of our model. The following computer is employed: 2 Intel Xeon E5-2609 v3 CPUs (each with 1.9GHz and 6 cores), 1 Titan X GPU (with 12G memory), and 128G RAM. For the NUS-WIDE dataset, the time of training our model is about three days. Moreover, the time of processing a test image is 0.01 second, i.e., our model can provide real-time annotation, which is crucial for real-world applications.

V. CONCLUSION

In this paper, we have proposed a multi-modal multi-scale deep learning model for large-scale image annotation. Different from the RNN-based models with implicit label quantity prediction, a regressor is directly added to our model

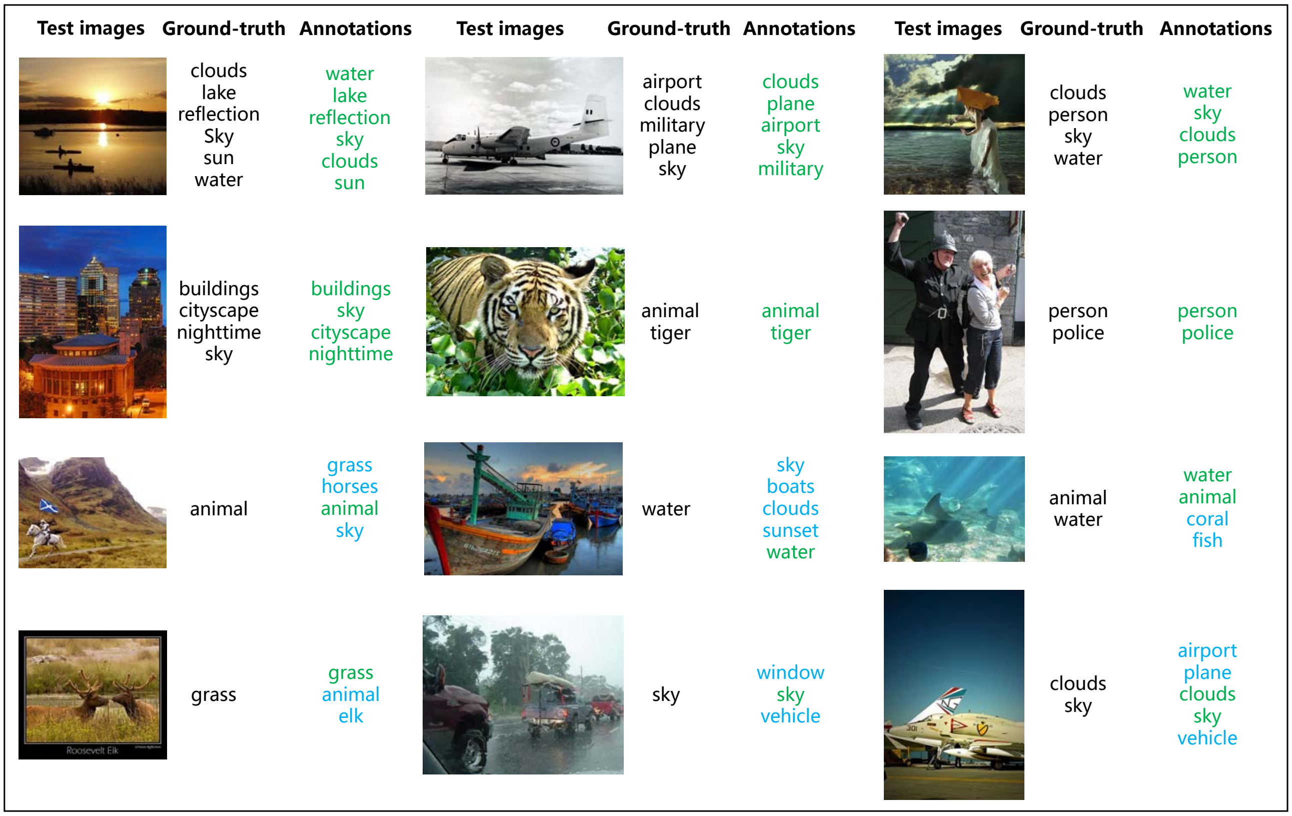


Fig. 6. Example results on NUS-WIDE obtained by our model. Annotations with green font are included in ground-truth class labels, while annotations with blue font are correctly tagged but not included in ground-truth class labels.

for explicit label quantity prediction. Extensive experiments are carried out to demonstrate that the proposed model outperforms the state-of-the-art methods. Moreover, each component of our model has also been shown to be effective in large-scale image annotation. A number of directions are worth further investigation. Firstly, more advancing CNN models can be adopted in our model. More efforts are needed to investigate how to improve the CNN models for large-scale image annotation. Secondly, other types of side information (e.g. group labels) can be fused in our model. Finally, the investigation of training our model in a singe process needs to be included in the ongoing work.

Acknowledgment

This work was partially supported by National Natural Science Foundation of China (61573363 and 61573026), 973 Program of China (2014CB340403 and 2015CB352502), the Fundamental Research Funds for the Central Universities and the Research Funds of Renmin University of China (15XNLQ01), and the Outstanding Innovative Talents Cultivation Funded Programs 2016 of Renmin University of China.

References

[1] A. Khotanzad and Y. H. Hong, “Invariant image recognition by Zernike moments,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 5, pp. 489–497, 2002.

[2] Y. L. Boureau, N. L. Roux, F. Bach, J. Ponce, and Y. Lecun, “Ask the locals: Multi-way local pooling for image recognition,” in *International Conference on Computer Vision*, 2011, pp. 2651–2658.

[3] Z. Lu, Y. Peng, and H. H. S. Ip, “Image categorization via robust pLSA,” *Pattern Recognition Letters*, vol. 31, no. 1, pp. 36–43, 2010.

[4] Z. Lu and H. S. Ip, “Image categorization with spatial mismatch kernels,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 397–404.

[5] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” in *Advances in Neural Information Processing Systems 25*, 2012, pp. 1097–1105.

[7] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[8] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.

[9] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, “Inception-v4, Inception-ResNet and the impact of residual connections on learning,” in *AAAI Conference on Artificial Intelligence*, 2017, pp. 4278–4284.

[10] A. Makadia, V. Pavlovic, and S. Kumar, “A new baseline for image annotation,” in *European Conference on Computer Vision*, 2008, pp. 316–329.

[11] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, “LabelMe: A database and web-based tool for image annotation,” *International Journal of Computer Vision*, vol. 77, no. 1–3, pp. 157–173, 2008.

[12] Z. Lu, H. H. S. Ip, and Y. Peng, “Contextual kernel and spectral methods for learning the semantics of images,” *IEEE Transactions on Image Processing*, vol. 20, no. 6, pp. 1739–1750, 2011.

[13] Z. Lu, P. Han, L. Wang, and J.-R. Wen, “Semantic sparse recoding of visual content for image applications,” *IEEE Transactions on Image Processing*, vol. 24, no. 1, pp. 176–188, 2015.

[14] X. Y. Jing, F. Wu, Z. Li, R. Hu, and D. Zhang, “Multi-label dictionary learning for image annotation,” *IEEE Transactions on Image Processing*, vol. 25, no. 6, pp. 2712–2725, 2016.

[15] G. Tsoumakas, I. Katakis, and D. Taniar, “Multi-label classification: An

overview," *International Journal of Data Warehousing & Mining*, vol. 3, no. 3, pp. 1–13, 2007.

[16] J. Read, B. Pfahringer, G. Holmes, and E. Frank, "Classifier chains for multi-label classification," *Machine Learning*, vol. 85, no. 3, pp. 333–359, 2011.

[17] J. Weston, S. Bengio, and N. Usunier, "Large scale image annotation: Learning to rank with joint word-image embeddings," *Machine Learning*, vol. 81, no. 1, pp. 21–35, 2010.

[18] X. Chen, Y. Mu, S. Yan, and T. S. Chua, "Efficient large-scale image annotation by probabilistic collaborative multi-label propagation," in *ACM International Conference on Multimedia*, 2010, pp. 35–44.

[19] D. Tsai, Y. Jing, Y. Liu, H. A. Rowley, S. Ioffe, and J. M. Rehg, "Large-scale image annotation using visual synset," in *IEEE International Conference on Computer Vision*, 2012, pp. 611–618.

[20] Z. Feng, R. Jin, and A. Jain, "Large-scale image annotation by efficient and robust kernel metric learning," in *IEEE International Conference on Computer Vision*, 2014, pp. 1609–1616.

[21] Y. LeCun, Y. Bengio, and G. E. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436–444, 2015.

[22] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.

[23] J. Johnson, L. Ballan, and L. Fei-Fei, "Love thy neighbors: Image annotation by exploiting image metadata," in *International Conference on Computer Vision*, 2015, pp. 4624–4632.

[24] F. Liu, T. Xiang, T. M. Hospedales, W. Yang, and C. Sun, "Semantic regularisation for recurrent image annotation," *arXiv preprint arXiv:1611.05490*, 2016.

[25] H. Hu, G.-T. Zhou, Z. Deng, Z. Liao, and G. Mori, "Learning structured inference neural networks with label relations," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2960–2968.

[26] J. Jin and H. Nakayama, "Annotation order matters: Recurrent image annotator for arbitrary length image tagging," in *International Conference on Pattern Recognition*, 2016, pp. 2452–2457.

[27] J. Wang, Y. Yang, J. Mao, Z. Huang, C. Huang, and W. Xu, "CNN-RNN: A unified framework for multi-label image classification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2285–2294.

[28] A. Graves, M. Liwicki, S. Fernandez, R. Bertolami, H. Bunke, and J. Schmidhuber, "A novel connectionist system for unconstrained handwriting recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 5, pp. 855–868, 2009.

[29] X. Li and X. Wu, "Constructing long short-term memory based deep recurrent neural networks for large vocabulary speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2015, pp. 4520–4524.

[30] K. Gregor, I. Danihelka, A. Graves, D. J. Rezende, and D. Wierstra, "DRAW: A recurrent neural network for image generation," in *International Conference on Machine Learning, pages=1462–1471, year=2015,*.

[31] Y. Gong, L. Wang, R. Guo, and S. Lazebnik, "Multi-scale orderless pooling of deep convolutional activation features," in *European Conference on Computer Vision*, 2014, pp. 392–407.

[32] S. Yang and D. Ramanan, "Multi-scale recognition with DAG-CNNs," in *International Conference on Computer Vision*, 2015, pp. 1215–1223.

[33] Z. Cai, Q. Fan, R. S. Feris, and N. Vasconcelos, "A unified multi-scale deep convolutional neural network for fast object detection," in *European Conference on Computer Vision*, 2016, pp. 354–370.

[34] J. Liu, Z. J. Zha, Q. I. Tian, D. Liu, T. Yao, Q. Ling, and T. Mei, "Multi-scale triplet CNN for person re-identification," in *ACM International Conference on Multimedia*, 2016, pp. 192–196.

[35] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng, "NUS-WIDE: A real-world web image database from National University of Singapore," in *ACM International Conference on Image and Video Retrieval*, 2009, p. 48.

[36] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: Lessons learned from the 2015 MSCOCO image captioning challenge," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 652–663, 2017.

[37] A. Ulges, M. Worring, and T. Breuel, "Learning visual contexts for image annotation from Flickr groups," *IEEE Transactions on Multimedia*, vol. 13, no. 2, pp. 330–341, 2011.

[38] K. Zhu, K. Zhu, and K. Zhu, "Automatic annotation of weakly-tagged social images on Flickr using latent topic discovery of multiple groups," in *Proceedings of the 2009 workshop on Ambient media computing*, 2009, pp. 83–88.

[39] Y. Gong, Y. Jia, T. Leung, A. Toshev, and S. Ioffe, "Deep convolutional ranking for multilabel image annotation," *arXiv preprint arXiv:1312.4894*, 2013.

[40] B. Dhingra, Z. Zhou, D. Fitzpatrick, M. Muehl, and W. W. Cohen, "Tweet2Vec: Character-based distributed representations for social media," in *Annual Meeting of the Association for Computational Linguistics*, 2016, pp. 269–274.

[41] W. Ling, C. Dyer, A. W. Black, and I. Trancoso, "Two/too simple adaptations of word2Vec for syntax problems," in *The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2015.

[42] X. Bai, F. Chen, and S. Zhan, "A new clustering model based on word2Vec mining on Sina Weibo users' tags," *International Journal of Grid Distribution Computing*, vol. 7, no. 3, pp. 44–48, 2014.

[43] Z. Lu, Z. Fu, T. Xiang, P. Han, L. Wang, and X. Gao, "Learning from weak and noisy labels for semantic segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 3, pp. 486–500, 2017.

[44] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, and M. Bernstein, "ImageNet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2014.

[45] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.