Fine-grained Image Classification by Visual-Semantic Embedding

Huapeng Xu[1], **Guilin Qi**[1,*], **Jingjing Li**[2], **Meng Wang**[3], **Kang Xu**[4] and **Huan Gao**[1]
[1] Southeast University, Nanjing, China
[2] University of Electronic Science and Technology of China, Chendu, China
[3] Xi'an Jiaotong University, Xi'an, China
[4] Nanjing University of Posts and Telecommunications, Nanjing, China
{xhp,gqi,gh}@seu.edu.cn, lijin117@yeah.net, wangmengsd@stu.xjtu.edu.cn, kxu@njupt.edu.cn

Abstract

This paper investigates a challenging problem, which is known as fine-grained image classification (FGIC). Different from conventional computer vision problems, FGIC suffers from the large intra-class diversities and subtle inter-class differences. Existing FGIC approaches are limited to explore only the visual information embedded in the images. In this paper, we present a novel approach which can use handy prior knowledge from either structured knowledge bases or unstructured text to facilitate FGIC. Specifically, we propose a visual-semantic embedding model which explores semantic embedding from knowledge bases and text, and further trains a novel end-to-end CNN framework to linearly map image features to a rich semantic embedding space. Experimental results on a challenging large-scale UCSD Bird-200-2011 dataset verify that our approach outperforms several state-of-the-art methods with significant advances.

1 Introduction

Fine-grained image classification aims to recognize sub-categories, such as identifying the species of birds [Wah *et al.*, 2011], under some basic-level categories. Different from the general-level object classification problem, fine-grained image classification is quite challenging due to the high degree of similarity among categories and the high degree of dissimilarity for a specific category caused by different poses, scales and so on (as shown in Figure 1). Most of the current approaches of FGIC attempt to learn discriminative visual representations [Lin *et al.*, 2015b; Jaderberg *et al.*, 2015] or try to localize various parts of the object [Zhang *et al.*, 2014; Huang *et al.*, 2016; Peng *et al.*, 2018] to capture fine-grained features for classification. These works focus on learning visual information with thousands of labeled images for each category. However, for human's recognition mechanism, when human beings recognize an object of an image, they not only focus on visual information but also consider some prior knowledge gained from their experience and text descriptions of the object. It is obvious that there is a latent correlation between visual and the external knowledge in human's brain. For example, when a man sees a bird of *black footed albatross* in the wild, he would recall some prior knowledge: *black footed albatross* has brown to black feathers with white around its eyes and bill (as illustrated in text column of Figure 2). Combining the visual and prior external information, human could able to classify it correctly.

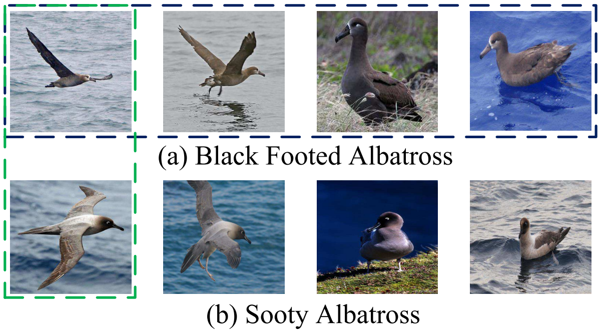


Figure 1: Illustration of the difficulty to discriminate large intra-class variance and small inter-class variance. Birds in each row are the same category with very large visual variance, but birds in each column are two different categories with a very similar visual aspect.

There are two kinds of prior external information, one is the text information and the other is knowledge base information. In the text context, class labels of images often have well-defined internal structure, where labels are more likely to co-occur with their related information (such as the feather color of *black footed albatross*, shown in the text column of Figure 2). Moreover, in the knowledge base context, class labels often contain multiple types and properties (as shown in the knowledge base column of Figure 2), and links among classes describe the relationship among them (in knowledge bases, a class label of an image is viewed as an entity). However, the existing approaches solve FGIC by artificially encoding image labels as sparse vectors by one-hot encoding, then train a deep network with softmax output layer. The problem of these approaches is that different labels are considered to be statistically independent, resulting in visual

[*] Corresponding author

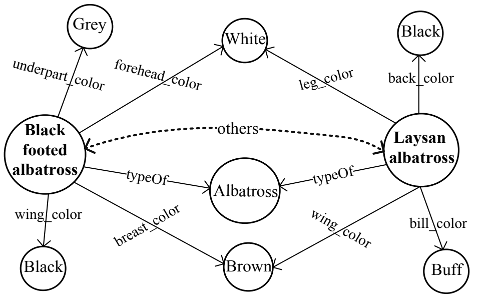
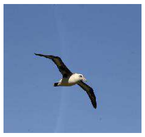
Category	Image	Text	Knowledge base
Black-footed albatross		Black-footed albatross is between two and three feet long with a large wingspan. It has brown to black feathers with white around its eyes and bill and has a large brown bill with a curved tip and black feet.	 Grey White Black underpart_color forehead_color leg_color back_color Black footed albatross others Laysan albatross typeOf Albatross typeOf wing_color breast_color wing_color bill_color Black Brown Buff
Laysan albatross		Laysan albatross has blackish-gray upperwing, back, and tail, and its head, underparts are white. It has a black smudge around the eye, and its underwing pattern with some having narrower black margins.	

Figure 2: Since subtle differences between fine-grained categories are often determined by the specific properties or parts of object, utilizing the fine properties of knowledge bases or text is significant for FGIC. Examples contain lots of discriminative information for recognizing *black-footed albatross* and *laysan albatross*, and properties of classes in knowledge bases shared characteristics of the two classes.

models that cannot leverage external knowledge to learn visual and semantic relationships. Suppose there are two class labels *black footed albatross* and *sooty albatross*. By one-hot encoding, these two labels will be encoded as 2-dimensional vectors: $a = [1, 0]$, $b = [0, 1]$. By applying traditional similarity measures (e.g., cosine similarity), the similarity of these two vectors is 0, that is to say none prior knowledge has been encoded to the class vectors.

To leverage the prior external knowledge, we propose a novel two-level convolutional neural network that uses handy prior knowledge from either structured knowledge bases or unstructured text to improve fine-grained image classification. We utilize a visual-semantic embedding framework to learn the relationship between classes and images. Specifically, our model is explicitly trained to project feature space of images into a rich semantic embedding space of classes in an end-to-end way, where the prior external knowledge is encoded into the embedding vectors of classes. Experimental results show that our method outperforms several state-of-the-art methods with significant advances.

2 Related Work

In this section, we review the current approaches in the field of FGIC and discuss some visual-semantic embedding approaches which use side information for visual tasks.

2.1 Fine-grained Image Classification

Over the past several years, many researchers have worked on exploring discriminative visual features or using part-based representation. These methods can be roughly categorized into three groups. For the first group, the works given in [Lin *et al.*, 2015b; Jaderberg *et al.*, 2015] attempt to get more discriminative visual representation by developing deep models, e.g., deep convolution neural network, for classifying fine-grained images. Bilinear CNN [Lin *et al.*, 2015b] considers a novel architecture that uses two separated CNNs to extract the visual part based on where the parts are and what the parts look like, which is slightly similar to our two-level CNN. Spatial transformer is introduced in [Jaderberg *et al.*, 2015], where a new differentiable module can be inserted into existing convolutional architectures to spatially transform feature maps without any extra training supervision. However, the subtle and local visual feature are particularly difficult to be captured. Therefore, some approaches [Zhang *et al.*, 2014; Huang *et al.*, 2016; Zhang *et al.*, 2016a] focus on part-based representation which tries to discovery classification criteria from object parts. However, these approaches need large amounts of human effort to annotate parts and bounding boxes. The last group tries to align the objects in different categories of images to reduce the influence of pose variations [Lin *et al.*, 2015a], or use object/parts spatial constraint to eliminate redundancy and enhances discrimination of selected parts [Peng *et al.*, 2018].

2.2 Visual-Semantic Embedding

The previous approaches of FGIC primarily focus on the visual information and ignore the external information. Many works use side information to solve other visual tasks [Marino *et al.*, 2017; Akata *et al.*, 2015; Frome *et al.*, 2013], and some of these works can be considered as visual-semantic embedding. SJE [Akata *et al.*, 2015] considers attribute-based image classification as a label-embedding problem for zero-shot learning. Specifically, a class is embedded in the space of attributes and learns a compatibility function between the image embedding and class embedding. Similarly, DeViSE [Frome *et al.*, 2013] finds that the semantic information can be exploited to make predictions about image classes, which uses image as input of CNN and Word2Vec [Mikolov *et al.*, 2013] representation as output embedding.

Different from all existing approaches, our method not only use external information, including knowledge bases and text, but also train it in an end-to-end manner. Attributes often are defined as discriminative properties, which have been shown very helpful to some visual recognition tasks [Akata *et al.*, 2015; Farhadi *et al.*, 2009]. We consider attributes as properties of classes into knowledge bases to enrich the embedding space.

3 The Proposed Model

In this section, we describe our two-level attention convolution neural network which jointly integrates semantic embedding from knowledge bases and text. Here, the core part of

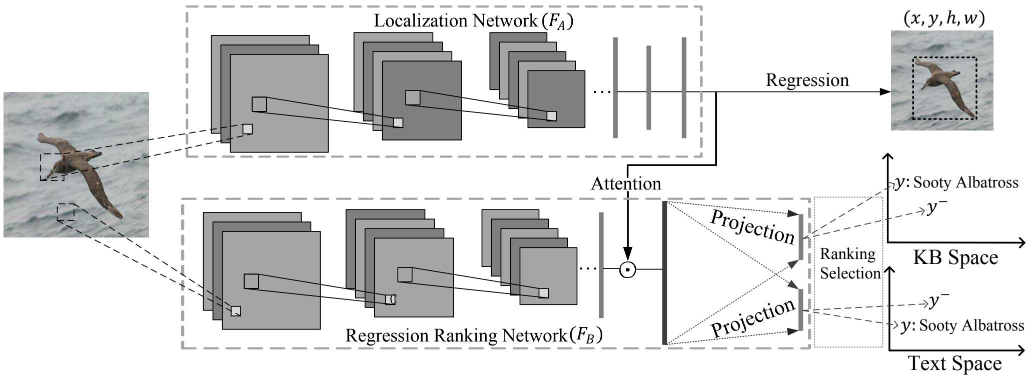


Figure 3: Our two-level convolutional neural network for FGIC. F_A aims to get the local feature of an object based on the detection mechanism and F_B works to linearly map visual feature of an image to the semantic embedding space (F_B embeds each image into nearby position of its corresponding label in the knowledge base embedding space and text embedding space).

our model is a visual-semantic mapping. Specifically, we use convolutional neural network to get image features as the visual embedding, and the visual embedding is linearly projected to its corresponding semantic embedding. We present a two-level CNN framework F_A and F_B. The overall model is summarized in Figure 3.

3.1 Problem Statement

Suppose that we have a labeled training dataset of images $\mathcal{X} = \{(x_i, y_i)\}(i = 1, ..., m)$, where each image is annotated with one of C fine-grained class labels, $\mathcal{Y} = \{y_1, y_2, ..., y_C\}$. The goal of our model is to learn a mapping function $f : \mathcal{X} \to \mathcal{Y}$ by minimizing the empirical loss which calculates the difference between the visual output of our model and the embedding of the true class y. Given a specific input image x, knowledge base embedding $\delta_1(y) \in \mathbb{R}^k$ of the corresponding class y and text embedding $\delta_2(y) \in \mathbb{R}^k$ of the corresponding class y, our model aims to maximize the posterior probability:

$$f(x, y) = argmax_{y \in Y} P(\delta_1(y), \delta_2(y)|x; \theta), \tag{1}$$

where θ is the learning parameter of our model. We use two different models TransR [Lin *et al.*, 2015c] and Word2Vec [Mikolov *et al.*, 2013] to get the embeddings $\delta_1(y)$ and $\delta_2(y)$ from textual corpus and knowledge bases respectively. Hence, $\delta_1(y)$ and $\delta_2(y)$ are conditionally independent. We reformulate Eq. 1 as follows:

$$f(x, y) = argmax_{y \in Y} \prod_{i \in 1,2} P(\delta_i(y)|x; \theta). \tag{2}$$

Eq. 2 is inspired from the work given in DeViSE [Frome *et al.*, 2013] and SJE [Akata *et al.*, 2015]. Within the visual-semantic embedding framework, DeViSE uses pairwise ranking objective function to explicitly map images into a rich semantic embedding space. SJE uses a compatibility function to map visual embedding of images and text embedding of classes while trains its model in a two-step way. Different from DeViSE and SJE, we integrate multiple domains (knowledge bases and text) into our model, and train our model in an end-to-end fashion.

3.2 Two-level Convolutional Neural Network

The subtle and local differences are crucial for distinguishing sub-categories and these differences often locate at the regions or parts of objects. Therefor, a two-stage framework is used by some previous works [Zhang *et al.*, 2014; Huang *et al.*, 2016]. The first stage is to localize the object or its discriminative parts based on a R-CNN framework [Girshick *et al.*, 2014], and the second stage is to extract visual features from the previous parts or the object. Bilinear CNN [Lin *et al.*, 2015b] uses two feature extractors based on CNN, where the first extractor emphasizes on the object identification and the second one focuses on the spatial location. The two CNN extractors consider pairwise interactions in a translationally invariant manner which is particularly suitable for fine-grained classification task. Based on Bilinear CNN and visual-semantic embedding work (e.g., DeViSE and SJE), we propose our two-level CNN.

Localization CNN: The first level of our model is to train a localization network (F_A) which aims to detect the bounding box of an object. The idea is inspired by region-based CNN [Zhang *et al.*, 2014] whose goal is to extract the candidate region proposals and check whether these proposals have the target objects. The feature of a region proposal is sensitive to the parts or the bounding of an object if the proposal is positive [Ouyang *et al.*, 2017]. For fine-grained classification, an image generally contains only one object which can be seen as a positive proposal. Thus, we use the raw image as a positive proposal and the output is similar to region-based CNNs. The learning target of F_A is the bounding box (x, y, h, w). Given an image, we formulate the objective of F_A as follows:

$$l_A = \sum_{i=1}^{4} (t_i - t_i')^2, \tag{3}$$

where t_i and t_i' $(i = 1, 2, 3, 4)$, for simplicity, are used to represent the true bounding box (x, y, h, w) and the output of F_A respectively.

Regression Ranking Network: The second level of our model is a regression ranking network (F_B) which tries to get the global visual feature of an image object. To integrate semantic embedding, F_B is trained jointly with two parallel fully connected (FC) layers without the softmax layer. The FC layers project the visual embedding of images (learned by deep CNN) to the semantic embedding of classes (learned by TransR or Word2Vec). The two parallel FC layers are called projection layers, where the first FC layer is to project the visual embedding to the text embedding (as DeViSE) of classes, and the second one is to project the visual embedding to the knowledge base embedding of classes. We use $M_1 \in \mathbb{R}^{d\times k}$ and $M_2 \in \mathbb{R}^{d\times k}$ as the parameters of projection layers and $v \in \mathbb{R}^d$ as visual embedding, where d is the size of visual embedding and k is the size of embedding vector of image classes.

To utilize the complementary information of the two different embeddings, F_B is trained to linearly project the visual embedding to the text embedding $\delta_1(y)$ and the knowledge base embedding $\delta_2(y)$ simultaneously when an x image is given. Because dot-product similarity measures the cosine of the angle between two vectors and euclidean distance gives the magnitude of the difference between the two vectors [Sidorov *et al.*, 2014], we propose a novel loss function which combines the dot-product similarity[1] and euclidean distance to measure the proximity between the projection results and a candidate class embeddings $\delta_1(y)$ and $\delta_2(y)$ of y. The formulation is defined as follows:

$$\pi(x,y) = \sum_{i=1}^{2}(1 - v^T M_i \delta_i(y) + |v^T M_i - \delta_i(y)|^2), \quad (4)$$

where $1 - v^T M_i \delta(y)_i$ is the negative dot-product similarity with margin 1 and the second part $|v^T M_i - \delta(y)_i|^2$ is the euclidean distance. For a fine-grained scenario, some classes are very difficult to distinguish the differences between them both in the visual and semantic space. To tackle this problem, the loss function of F_B is defined as follows:

$$l_B = \pi(x,y) - \pi(x,y^-), \quad (5)$$

where y^- is selected by a ranking formula:

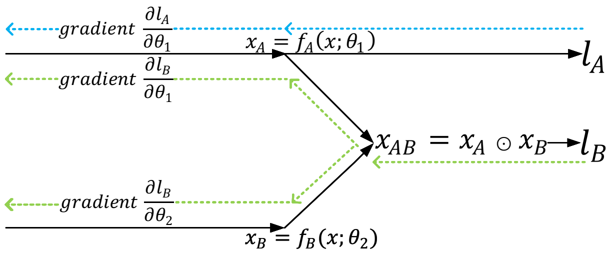
$$y^- = argmin_{(y'\in\mathcal{Y}, y'\neq y)}\pi(x,y'). \quad (6)$$

These functions (i.e, Eq. 6 and Eq. 5) are designed such that, for a given image, the distance of the visual output of F_B is encouraged to be closer to the embedding of the positive class y. Furthermore, the distances of the visual output of F_B and the embeddings of all negative classes are encouraged to be maximized, since the embedding of the selected y^- has the minimal distance with the visual output of F_B.

3.3 Parameter Learning

We model localization network's features and the regression ranking network's features by Hadamard product (elementwise multiplication) as in shown Figure 3. By learning the bounding box in F_A and the class embedding of object in F_B simultaneously, our model can be seen as a multi-task learning with the objective loss as follows:

$$L(x,y) = \alpha * l_A + l_B, \quad (7)$$



gradient $\frac{\partial l_A}{\partial \theta_1}$
gradient $\frac{\partial l_B}{\partial \theta_1}$
gradient $\frac{\partial l_B}{\partial \theta_2}$
$x_A = f_A(x;\theta_1)$
l_A
$x_{AB} = x_A \odot x_B$
l_B
$x_B = f_B(x;\theta_2)$

Figure 4: Gradients influence of our two-level CNN.

where α is a hyperparameter. By adding the two loss functions l_A and l_B in an overall architecture, our model can be trained by back-propagation in an end-to-end form. Suppose the learned features of localization network and regression ranking network are fixed-size vectors $x_A = f_A(x;\theta_1)$ and $x_B = f_B(x;\theta_2)$, where θ_1 and θ_2 are the parameters of F_A, F_B respectively. The output of Hadamard product is:

$$x_{AB} = x_A \odot x_B. \quad (8)$$

The gradients of l_A only back-propagate F_A by chain rule, while the gradients of l_B back-propagate to F_A and F_B. Let $\frac{\partial L(x,y)}{\partial \theta}$ be the gradient of the loss w.r.t. θ, we get the gradients:

$$\begin{aligned}\frac{\partial L(x,y)}{\partial \theta_1} &= \alpha * \frac{\partial l_A}{\partial \theta_1} + \frac{\partial l_B}{\partial \theta_1}\\ &= \alpha * \frac{\partial l_A}{\partial f_A}\frac{\partial f_A}{\partial \theta_1} + \frac{\partial l_B}{\partial x_{AB}}\frac{\partial x_{AB}}{\partial f_A}\frac{\partial f_A}{\partial \theta_1} \\ &= \alpha * \frac{\partial l_A}{\partial f_A}\frac{\partial f_A}{\partial \theta_1} + x_B\frac{\partial l_B}{\partial x_{AB}}\frac{\partial f_A}{\partial \theta_1},\end{aligned} \quad (9)$$

$$\begin{aligned}\frac{\partial L(x,y)}{\partial \theta_2} &= \alpha * \frac{\partial l_A}{\partial \theta_2} + \frac{\partial l_B}{\partial \theta_2}\\ &= \alpha * \frac{\partial l_A}{\partial f_B}\frac{\partial f_B}{\partial \theta_2} + \frac{\partial l_B}{\partial x_{AB}}\frac{\partial x_{AB}}{\partial f_B}\frac{\partial f_B}{\partial \theta_2} \\ &= x_A\frac{\partial l_B}{\partial f_B}\frac{\partial f_B}{\partial \theta_2}.\end{aligned} \quad (10)$$

A simple illustration of parameter learning process in our model is shown in Figure 4. The gradients of F_B back-propagate to the total network (as shown in Eq. 9), so the classification error makes an interaction learning between F_A and F_B. We observe the learned features x_A of F_A are usually sensitive to discriminative parts, such as head and tail. Thus, the features learned by F_A can be seen as attention information to F_B, which shows that the discriminative features embedded in x_B of F_B will be given more attention by filtering less import information via x_A of F_A, such as the background of an image[Wang *et al.*, 2016].

4 Semantic Embedding of Image Class

Semantic embedding indicates latent representation of classes in knowledge bases or text . In this section, we will first clarify the embeddings used in our model, and introduce how we extract embedding representation of classes from knowledge bases and text.

[1] Dot product similarity is an equivalent similarity measurement as cosine similarity when normalize vectors in unit form.

Method	Train Anno	Train BBox	Test Anno	Test BBox	Accuracy
PB R-CNN [Zhang *et al.*, 2014]	✓	✓	✓	✓	82.02%
PS R-CNN [Huang *et al.*, 2016]	✓	✓		✓	76.2%
SPDA-CNN [Zhang *et al.*, 2016a]	✓	✓		✓	85.14%
ST CNN [Jaderberg *et al.*, 2015]					84.1%
Bilinear CNN [Lin *et al.*, 2015b]					84.1%
PDFS [Zhang *et al.*, 2016b]					84.5%
CVL [He and Peng, 2017]					85.55%
Our T-CNN					**86.2%**
Our T-CNN average		✓			**86.5%**
Ensemble T-CNN		✓			**87.3%**

Table 1: Experimental results on Caltech-UCSD Bird dataset, where BBox refers to bounding box and Anno refers to part annotations.

4.1 Knowledge Base Embedding

Knowledge base (KB) is a repository which stores complex structured information about things of our world. KB contains multiple types of entities with their properties and links among entities, and the links describe the relationship of entities. To fully represent entities in KB as low dimensional vectors, a novel KB embedding model called TransR [Lin *et al.*, 2015c] is applied to get the embedding representation of entities. In TransR, KB is a set of triples, as the form $(\boldsymbol{h},\boldsymbol{r},\boldsymbol{t})$, where $\boldsymbol{h}$, $\boldsymbol{t}$, and $\boldsymbol{r}$ indicate head entity, tail entity, and relationship between the two entities respectively. For each triple $(\boldsymbol{h},\boldsymbol{r},\boldsymbol{t})$, entities and relationship are first embedded into continuous vectors $h \in \mathbb{R}^{d'}$, $t \in \mathbb{R}^{d'}$ and $r \in \mathbb{R}^{k}$, then entity vectors are projected into the r-relation space as $h_r \in \mathbb{R}^k$ and $t_r \in \mathbb{R}^k$ using a relation-specific matrix $M_r \in \mathbb{R}^{d' \times k}$. $h_r = hM_r, t_r = hM_r$. The objective of TransR is a translation based function: $f_r(h,t) = ||h_r + r - t_r||_2^2$, which makes the learned vectors of entities to reserve latent structured knowledge.

Attribute-based KB. In general KB (e.g., DBpedia), it has a huge coverage of knowledge across domains, but it has incomplete knowledge in specific domains. A specific domain in general KB lacks of lots of fine-grained properties, i.e., semantic attributes of fine-grained objects [Akata *et al.*, 2015]. The attributes provide a way to describe such fine-grained concepts. For example, the wing color of *black footed albatross* is grey, which is a very significant part to distinguish *black footed albatross*. Suppose there exist C classes of images, $\mathcal{Y} = \{y_1, y_2, ..., y_C\}$ and E attributes $\mathcal{A} = \{a_1, a_2, ..., a_E\}$, to make the embedding vector of classes to capture those fine-grained properties, we construct domain-specific attribute knowledge into the general KB, i.e., we construct a triple (y, *has_property_of*, a_i) for each attribute a_i of each class y .

4.2 Text Embedding

Word2Vec [Mikolov *et al.*, 2013] is a shallow neural network that is trained to reconstruct linguistic contexts of words. The embedding of a class term from Word2Vec can capture semantically-meaningful properties of the class since property words often couple with the context of a class word.

Fine-tuning Word2Vec. In Word2Vec, the embeddings of words are affected by their statistical contexts [Mikolov *et al.*, 2013], so it is difficult to accurately learn the embedding vectors of infrequent words. For a fine-grained scenario, the class labels often belong to the infrequent words in general domain text, such as $Wikipedia$. In order to learn high-quality representation of class labels, we first train Word2Vec in $Wikipedia$ corpus as the initialization of Word2Vec, then fine-tune Word2Vec on a domain-specific corpora, where the domain-specific corpora is extracted from $Wikipedia$.

5 Experiments

In this section, we present the experimental settings and show experimental results of our proposed model on the widely-used benchmark Caltech-UCSD Bird-200-2011 [Wah *et al.*, 2011]. The dataset contains 200 bird categories with 312 attributes. We choose DBpedia [Lehmann *et al.*, 2015] (KB) and English-language $Wikipedia$ (text) from 06.01.2016 as external knowledge. Word2Vec and TransR (described in Section 4) are used to get the class embedding.

5.1 Experiment Setting

In our experiment, the regression ranking network of our model is forked from existing CNN architectures with fine tuning (e.g., AlexNet [Krizhevsky *et al.*, 2012], VGG [Simonyan and Zisserman, 2014], GoogleNet [Szegedy *et al.*, 2016], ResNet [He *et al.*, 2016]). In addition, two projection layers are added to the regression ranking network after the last fully connected layer. We first resize all images to 224×224 pixels and then get image feature vectors from F_B. The projection layers of F_B project the visual features to the embedding of classes. The localization network is based on AlexNet which is also trained as regression to get the bounding box of an object. We train our model using stochastic gradient descent with mini-batches 40 and learning rate 0.0015. The hyperparameter α of Eq. 7 is set to be 0.85 with cross-validation . To train a fast convergent network, batch-normalization [Ioffe and Szegedy, 2015] layer is used after each convolutional layer. Random dropout tricks [Srivastava *et al.*, 2014] are used in the fully connected layers for alleviating overfitting. Moreover, all parameters of convolutional layers of F_B are pre-trained on the ImageNet and fine-tuned on the Caltech-UCSD data.

We use two different embeddings of classes based on Word2Vec and TransR. For Word2Vec embedding, we first train Word2Vec on $Wikipedia$ with one pass through the

Method	Accuracy
ResNet+Embedding	85.78%
AlexNet+ResNet+Embedding	86.2%
AlexNet+ResNet+Embedding+BBox	87.0%

Table 2: Analysis of different variants of our model on Caltech-UCSD Bird.

corpus, then fine-tune Word2Vec on a domain-specific corpus extracted from $Wikipedia$ articles which are related with Caltech-UCSD Birds. For TransR embedding, we combine DBpedia with Attribute-based KB (described in 4.1) as the knowledge base and train TransR with the same setting in [Lin *et al.*, 2015c] to get the embedding.

5.2 Classification Result and Comparison

The results of our proposed two-level CNN and the state-of-art methods are presented in Table 1. We list bounding box and part annotation for fair comparison. From the results, we can see that part-based methods[Zhang *et al.*, 2014; 2016a], which use both annotation and bounding box get the best result of 82.02% and 85.14%, respectively. Our model, however, can achieve an average accuracy about 86.5% with only bounding box needed in the training step. This (our T-CNN average row in Table 1) verifies our proposition that the external information from KB and text is helpful for fine-grained classification without any part annotations. Compared with Bilinear CNN [Lin *et al.*, 2015b] and PDFS [Zhang *et al.*, 2016b] which are free of any bounding box or part annotation at both training and testing stages, we change our model with only the second loss l_B, in other words, F_A is used for feature extraction which is similar with Bilinear CNN. We achieve an accuracy (Our T-CNN row in Table 1) 86.2% compared with 84.1% of Bilinear CNN and 84.5% of PDFS. It verities the effectiveness of our two-level CNN, which jointly integrates visual and semantic embedding to exploit the correction between visual and external knowledge. Furthermore, we compare with the latest state-of-art CVL [He and Peng, 2017] which utilizes the prior language descriptions in a language learning stream to point out the discriminative parts of images, our model performs about 1% better than CVL. This also verities that the prior external knowledge can be used via the visual-embedding way. We ensemble the results of different CNN-based architectures, and get the accuracy of 87.3%. All in all, the results clearly demonstrate the effectiveness of our two-level CNN-based model.

5.3 Model Analysis

We perform detailed analysis by comparing different variants of our model: (1) Only regression ranking network as F_B with semantic embedding; (2) Two-level CNN model without using bounding box; (3) Our full two-level CNN model with bounding box at training stage. The results are shown in Table 2. By leveraging semantic embedding from external knowledge, we get the highest result 87.0% when the most powerful feature extractors (AlexNet+ResNet) are used. By comparing the setting of (1) and (2), we demonstrate the effectiveness of localization network. In particular, the feature learned by localization network is attention information to our regression ranking network.

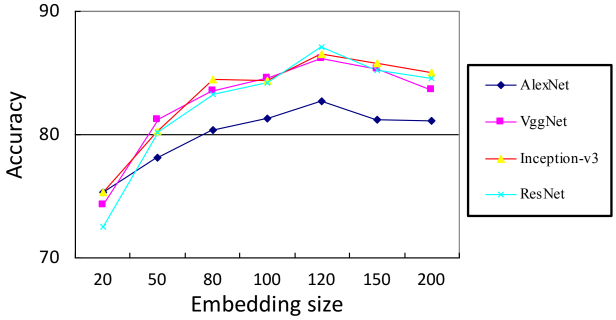


Figure 5: Results of different embedding size with different CNN-based architectures of our model on Caltech-UCSD Bird dataset.

To evaluate the impact of dimension of embedding vector to the classification result, we train Word2Vec and TransR with various embedding dimensions, ranging from 20-D to 200-D. We also test different CNN-based architectures of our regression ranking network. Experimental results show that the best size of dimension is 120 (as shown in Figure 5), with the size we can achieve an accuracy more than 86.0% on average. Specifically, dimension size 120 of embedding vector can make the representation be more discriminative for FGIC.

6 Conclusion

In this paper, we proposed a novel two-level CNN regression model as a way of efficiently leveraging external knowledge to improve FGIC. In particular, we observed that the implicit structured information of KB and unstructured information of text can be embedded into semantic embedding vectors, then our method utilized the semantic embedding in a visual-semantic embedding framework. Moreover, by leveraging external knowledge, our model was more interpretable and similar with human recognition mechanism. One important advantage of our method was that our two-level CNN could reinforce each other, which led to capturing better discriminative features for fine-grained classification, as the two networks were trained in an end-to-end fashion to have pairwise interaction. The experimental results on a widely used Caltech-UCSD Bird dataset shown that our proposed model can outperform state-of-the-art methods. In the future, we would consider deep symbolic reasoning on the external knowledge into our work to make our model more reasonable and interpretable as the human recognition mechanism.

Acknowledgments

This work is mainly supported by the National Key R&D Program of China under Grant no. 2017YFB1002801 and the National Natural Science Foundation of China under Grant no. U1736204. JingJing Li is supported by the National Postdoctoral Program for Innovative Talents under Grant no. BX201700045 and China Postdoctoral Science Foundation under Grant no. 2017M623006.

References

[Akata *et al.*, 2015] Zeynep Akata, Scott E. Reed, Daniel Walter, Honglak Lee, and Bernt Schiele. Evaluation of output embeddings for fine-grained image classification. In *CVPR*, pages 2927–2936, 2015.

[Farhadi *et al.*, 2009] Ali Farhadi, Ian Endres, Derek Hoiem, and David A. Forsyth. Describing objects by their attributes. In *CVPR*, pages 1778–1785, 2009.

[Frome *et al.*, 2013] Andrea Frome, Gregory S. Corrado, Jonathon Shlens, Samy Bengio, Jeffrey Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. In *NIPS*, pages 2121–2129, 2013.

[Girshick *et al.*, 2014] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, pages 580–587, 2014.

[He and Peng, 2017] Xiangteng He and Yuxin Peng. Fine-grained image classification via combining vision and language. In *CVPR*, pages 7332–7340, 2017.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[Huang *et al.*, 2016] Shaoli Huang, Zhe Xu, Dacheng Tao, and Ya Zhang. Part-stacked CNN for fine-grained visual categorization. In *CVPR*, pages 1173–1182, 2016.

[Ioffe and Szegedy, 2015] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, pages 448–456, 2015.

[Jaderberg *et al.*, 2015] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. In *NIPS*, pages 2017–2025, 2015.

[Krizhevsky *et al.*, 2012] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1106–1114, 2012.

[Lehmann *et al.*, 2015] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, Sören Auer, and Christian Bizer. Dbpedia - A large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web Journal*, pages 167–195, 2015.

[Lin *et al.*, 2015a] Di Lin, Xiaoyong Shen, Cewu Lu, and Jiaya Jia. Deep LAC: deep localization, alignment and classification for fine-grained recognition. In *CVPR*, pages 1666–1674, 2015.

[Lin *et al.*, 2015b] Tsung-Yu Lin, Aruni.R Chowdhury, and Subhransu Maji. Bilinear CNN models for fine-grained visual recognition. In *ICCV*, pages 1449–1457, 2015.

[Lin *et al.*, 2015c] Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. Learning entity and relation embeddings for knowledge graph completion. In *AAAI*, pages 2181–2187, 2015.

[Marino *et al.*, 2017] Kenneth Marino, Ruslan Salakhutdinov, and Abhinav Gupta. The more you know: Using knowledge graphs for image classification. In *CVPR*, pages 20–28, 2017.

[Mikolov *et al.*, 2013] Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, pages 3111–3119, 2013.

[Ouyang *et al.*, 2017] Wanli Ouyang, Xingyu Zeng, Xiaogang Wang, Shi Qiu, Ping Luo, Yonglong Tian, Hongsheng Li, Shuo Yang, Zhe Wang, Hongyang Li, et al. Deepid-net: Object detection with deformable part based convolutional neural networks. *IEEE TPAMI*, pages 1320–1334, 2017.

[Peng *et al.*, 2018] Yuxin Peng, Xiangteng He, and Junjie Zhao. Object-part attention model for fine-grained image classification. *IEEE Trans. Image Processing*, pages 1487–1500, 2018.

[Sidorov *et al.*, 2014] Grigori Sidorov, Alexander Gelbukh, Helena G.Adorno, and David Pinto. Soft similarity and soft cosine measure: Similarity of features in vector space model. *Computación y Sistemas*, pages 491–504, 2014.

[Simonyan and Zisserman, 2014] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR, abs/1409.1556*, 2014.

[Srivastava *et al.*, 2014] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, pages 1929–1958, 2014.

[Szegedy *et al.*, 2016] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, pages 2818–2826, 2016.

[Wah *et al.*, 2011] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

[Wang *et al.*, 2016] Sen Wang, Pingbo Pan, Guodong Long, Weitong Chen, Xue Li, and Quan Z. Sheng. Compact representation for large-scale unconstrained video analysis. *World Wide Web*, pages 231–246, 2016.

[Zhang *et al.*, 2014] Ning Zhang, Jeff Donahue, Ross B. Girshick, and Trevor Darrell. Part-based r-cnns for fine-grained category detection. In *ECCV*, pages 834–849, 2014.

[Zhang *et al.*, 2016a] Han Zhang, Tao Xu, Mohamed Elhoseiny, Xiaolei Huang, Shaoting Zhang, Ahmed Elgammal, and Dimitris Metaxas. Spda-cnn: Unifying semantic part detection and abstraction for fine-grained recognition. In *CVPR*, pages 1143–1152, 2016.

[Zhang *et al.*, 2016b] Xiaopeng Zhang, Hongkai Xiong, Wengang Zhou, Weiyao Lin, and Qi Tian. Picking deep filter responses for fine-grained image recognition. In *CVPR*, pages 1134–1142, 2016.