



A Unified Modular Framework with Deep Graph Convolutional Networks for Multi-label Image Recognition

Qifan Lin^{1,2} , Zhaoliang Chen^{1,2} , Shiping Wang^{1,2} ,
and Wenzhong Guo^{1,2}

¹ College of Mathematics and Computer Science, Fuzhou University,
Fuzhou, Fujian, China

guowenzhong@fzu.edu.cn

² Fujian Provincial Key Laboratory of Network Computing and Intelligent
Information Processing, Fuzhou University, Fuzhou, Fujian, China

Abstract. With the rapid development of handheld photographic devices, a large number of unlabeled images have been uploaded to the Internet. In order to retrieve these images, image recognition techniques have become particularly important. As there is often more than one object in a picture, multi-label image annotation techniques are of practical interest. To enhance its performance by fully exploiting the interrelationships between labels, we propose a unified modular framework with deep graph convolutional networks (MDGCN). It consists of two modules for extracting image features and label semantic respectively, after which the features are fused to obtain the final recognition results. With classical multi-label soft-margin loss, our model can be trained in an end-to-end schema. It is important to note that a deep graph convolutional network is used in our framework to learn semantic associations. Moreover, a special normalization method is employed to strengthen its own connection and avoid features from disappearing in the deep graph network propagation. The results of experiments on two multi-label image classification benchmark datasets show that our framework has advanced performance compared to the state-of-the-art methods.

Keywords: Multi-label image recognition · Convolutional neural networks · Graph convolutional networks · Feature extraction

1 Introduction

Every time we press the shutter, we get a picture. Nowadays, anyone can easily takes a picture using a mobile phone, which has led to an explosion in the number of images circulating on the Internet. How a computer performs image recognition automatically is a significant task for computer vision researchers. Since there is often more than one subject in the images we take, the task of multi-label image recognition is of great practical importance. And it has been

used in many areas, for example, in medical image recognition analysis [14] to help doctors improve diagnostic efficiency and in the image retrieval field [16] to effectively reduce the amount of manually annotated data.

The basis of image recognition is image feature extraction. In single-label image classification, deep convolutional neural networks have shown advanced performance. Once ResNet [10] was proposed in 2016, it has been gradually replacing the VGG [17] in many practical scenarios. In the following year, Kaiming He’s team proposed an upgraded version of the deep residual network called ResNeXt [22]. It can be used for image feature extraction. By simply appending a linear layer of the network extracting feature dimensions to the categories with threshold control, it can perform multi-label image classification tasks.

The feature extraction network utilizes the images from the training set, but does not fully exploit the semantic features of the given labels. Therefore, CNN-RNN [19], SRN [24], RNN-Attention [20], ML-GCN [4] and other frameworks that combine deep neural networks with labeled semantic information have been proposed one after another.

How to depict the relationship between labels is an important issue, and it is worth noting that graph networks have this advantage. Graph convolutional network (GCN) was proposed by Kipf and Welling [11] in 2017. It was initially used in semi-supervised image classification and achieved advanced performance. A number of researchers have invested in studying GCN and have proposed many methods based on it, such as Cluster-GCN [5] and AdaGCN [18].

In this paper, we introduce a unified modular framework with deep graph convolutional networks. It contains two main modules, one module adopts convolutional neural networks to extract image features, and the other module employs deep graph convolutional networks to extract label semantic features. The final multi-label image classification results will be obtained by fusing the features of the two aforementioned modules. The main contributions of this paper can be concluded as follows:

- An end-to-end trainable framework is proposed for multi-label image recognition tasks, where both image representation and label semantic features can be extracted and fully used.
- A deep graph convolutional network is employed to extract semantic features, where the adjacency matrix is regularized so that the contribution of nearby nodes is greater than that of the more distant nodes.
- Experiments are carried out on two multi-label image benchmark datasets, MS-COCO and VOC 2007, and the experimental results show the competitive performance of our approach compared to the state-of-the-art methods.

2 Related Work

In the past fifteen years, the performance of image classification has witnessed rapid progress. The reason for this is due to the establishment of large-scale hand-labeled datasets such as ImageNet [6], Microsoft COCO [13] and PASCAL VOC [7], and the fast development of deep convolutional networks such as ResNet

[10]. Many studies have been devoted to extending deep convolutional networks to the field of multi-label image recognition.

A straightforward way to solve the multi-label image classification problem is to train multiple binary classifiers. However, this is not realistic as the number of binary classifiers required for it is exponential, requiring 2^{20} binary classifiers for a dataset of just twenty classes. It can also be performed by training a classifier for each class and controlling the classification results using thresholding or ranking [9]. However, many researchers believe that what limits its performance improvement is not considering the dependencies between labels.

In order to exploit label dependencies, some studies have been carried out. Wang et al. [19] used recurrent neural networks to learn a joint low-dimensional image-label embedding to model the semantic relevance between images and labels. Zhu et al. [24] proposed the Spatial Regularization Net to generate attention maps for all labels and capture the underlying relations between them via learnable convolutions. Wang et al. [20] designed a special recurrent memorized-attention module, composed of a spatial transformer and an LSTM network, to take full advantage of the label distributions.

Different from these methods mentioned above, the researchers found that the graph structure could be more effective in modeling label co-occurrence relationships. Chen et al. [4] proposed a framework for combining multi-label image recognition with graph convolutional networks, using the graph network to map label representations into a series of inter-dependent object classifiers. Chen et al. [3] proposed a graph neural network semantic extraction structure called SSGRL, which extracts specific semantic relationships between labels through multiple graph neural network layers.

Unlike previous work, our framework MDGCN focuses on using deep graph convolutional networks to obtain better semantic features. In addition, a normalization method is used to consider that the information should be enhanced for nodes nearby rather than distant nodes.

3 Proposed Method

Given the image set I , the label set L , multi-label image recognition is supposed to predict the image labels \hat{y} . In this paper, we propose our MDGCN framework, as shown in Fig. 1. We design an image feature extraction module to perform representation learning in image set. In label set L , a label semantic extraction module is designed to fully explore the label semantic information and co-occurrence relationship between labels. Then, the final classification result \hat{y} is obtained by feature fusion.

3.1 Image Feature Extraction Module

We consider an RGB image as a tensor with the dimension of $w \times h \times 3$. Original input would result in a model with too large parameters and slow down the training process, attributed to which we designed the image feature extraction

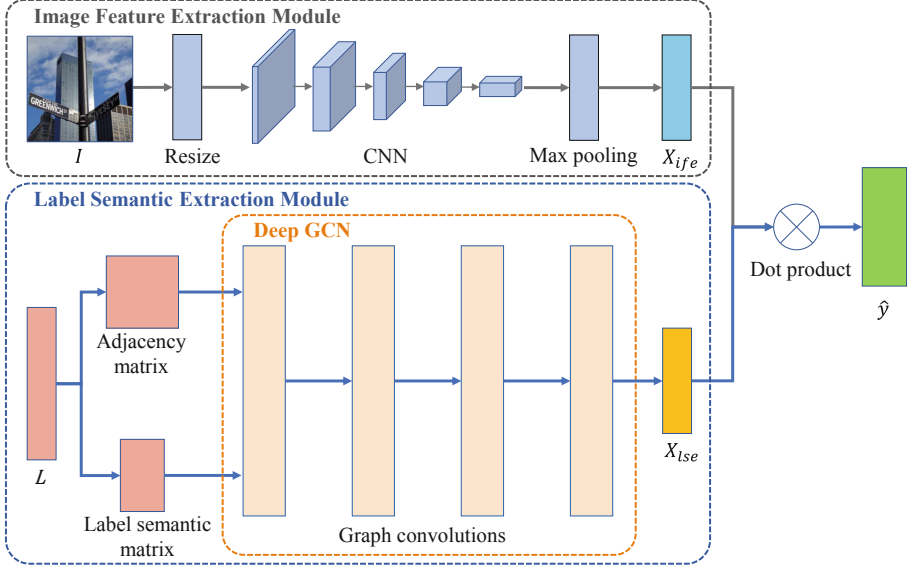


Fig. 1. Overall framework of our MDGCN model for multi-label image recognition.

module (IFEM) to perform feature extraction on the original input. In this module, we use the convolutional neural networks (CNN) which is commonly used in representation learning. It can be summarized as

$$F_{image} = f_{cnn}(f_{resize}(I)), \quad (1)$$

where $I \in \mathbb{R}^{h \times w \times 3}$ represents the set of input images, $f_{resize}(\cdot)$ denotes a size scaling function, and F_{image} is the features obtained after the convolutional layers of CNN. We can adopt any kind of CNN in this part. To adapt its output dimension to the entire framework, we employ a max pooling function. And it enables the model to have better generalization ability. The formula is expressed as follow

$$X_{ife} = f_{mp}(F_{image}), \quad (2)$$

where $f_{mp}(\cdot)$ denotes a max pooling function, and $X_{ife} \in \mathbb{R}^{n \times s}$ indicates the image features obtained by the image feature extraction module. The input image set I is transformed into a feature representation with dimension s , according to Eq. (1) and Eq. (2).

3.2 Label Semantic Extraction Module

Our intention in designing the label semantic extraction module (LSEM) is to make full use of the given label information to achieve better multi-label image classification results. The first question we think about is what information we can get from the labels of the training set and how we can make full use of

this information. Previous work has given us some inspirations. We can obtain the meaning of words themselves by learning word embedding and calculate the label co-occurrence probability through statistical methods.

Semantic Vector of LSEM. The training set labels, such as dog, aircraft, cars, are often treated as completely different category markers in classification methods. However, the semantic meaning of the label is overlooked. We consider that aircraft and cars are both vehicles, so the distance between them should be smaller than that between dogs and aircraft. In order to transform semantic labels into vectors with smaller Euclidean distances for labels with similar semantics, we use a word-to-vector method. The definition is as follows

$$X_{lsv} = f_{wsv}(L_t), \quad (3)$$

where $X_{lsv} \in \mathbb{R}^{C \times X_{gci}}$ denotes the label semantic matrix, which is obtained by passing the training set label $L_t \in \mathbb{R}^C$ through the semantic-to-vector transformation function $f_{wsv}(\cdot)$.

Correlation Matrix of LSEM. How the dependencies between labels are described is another key point for LSEM to be effective. As shown in Fig. 2, we employ conditional probabilities to construct label co-occurrence probability. For example, $P(C_a | C_b)$ denotes the co-occurrence probability of class a , given the co-occurrence of class b . Therefore, $P(C_a | C_b)$ is not equal to $P(C_b | C_a)$, which constitutes a directed graph.

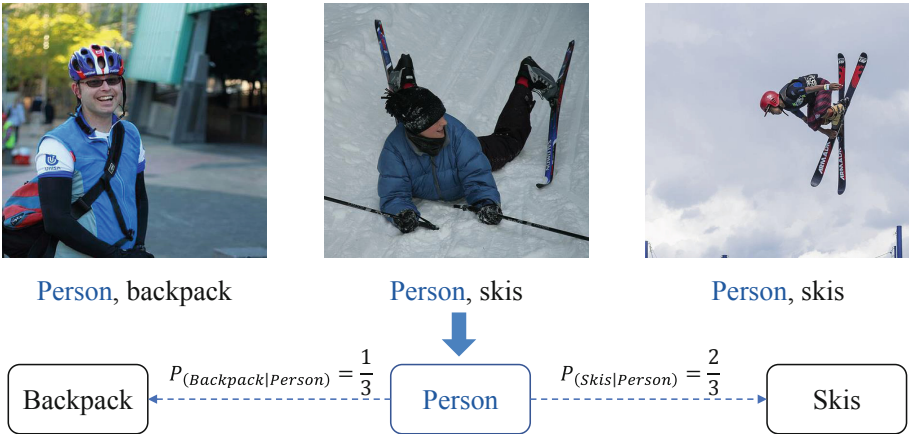


Fig. 2. Schematic diagram of correlation matrix construction. It shows the co-occurrence probability between “person” and the other two labels.

We denote the label co-occurrence matrix by P . Therefore, we need to count the number of times that the label appears together. The formula for calculating the co-occurrence probability P of a label is as follows

$$P_i = T_i/N_i, \quad (4)$$

where $T \in \mathbb{R}^{C \times C}$ indicates the number of times that the label co-occurs, and N_i is the number of appearances of the i -th tag in the training set.

Deep Graph Convolutional Networks. The original recursive formula for a graph convolutional network is expressed as follows

$$X^{(l+1)} = f\left(A, X^{(l)}\right), \quad (5)$$

where $X^{(l)}$ denotes the l -th layer of GCN, and A is the adjacency matrix used to describe neighborhood relationships.

As the input matrix increases, the performance of the original two-layer GCN exhibits some limitations. Common approaches are to expand the parameters of each layer of the network or to deepen the network. Therefore, we have designed a deep graph convolutional network (DGCN).

We define a DGCN as $f(A, X^{(1)})$ where A substitutes the adjacency matrix and the input to first layer is $X^{(1)} = X_{lsv}$. We improve the original $\tilde{A} = (A + I)$ by adding an identity and normalizing, that is,

$$\tilde{A} = (D + I)^{-1}(A + I), \quad (6)$$

where $A = P$ and $D_{ii} = \sum_j A_{ij}$, then we can define the recursive formula of DGCN as follows

$$X^{(l+1)} = \delta\left(\left(\tilde{A} + \text{diag}(\tilde{A})\right) X^{(l)} W^{(l)}\right), \quad (7)$$

where $\delta(\cdot)$ is an activation function. Using Eq. (7), we can get the final output of the label semantic extraction module X_{lse} .

3.3 Prediction Results and Training Scheme

After the two modules described above, we can obtain the final prediction results \hat{y} according to the following equation:

$$\hat{y} = X_{ife} \otimes X_{lse}, \quad (8)$$

where \otimes denotes dot product, X_{ife} is the output of image feature extraction module, and X_{lse} is the output of label semantic extraction module.

In this work, MDGCN is trained with a multi-label soft-margin loss, which creates a criterion that optimizes a multi-label one-versus-all loss between predicted result \hat{y} and the ground truth label y based on max-entropy. The loss function equation is defined as follows

$$\mathcal{L} = -\frac{1}{C} \sum_{c=1}^C y_c \log(\sigma(\hat{y}_c)) + (1 - y_c) \log(1 - \sigma(\hat{y}_c)), \quad (9)$$

where C is the number of classes, and $\sigma(\cdot)$ denotes a sigmoid function. Algorithm 1 illustrates an epoch of the proposed MDGCN in detail.

Algorithm 1. MDGCN**Input:** Image set I , label set L , and depth of the graph convolutional network l_{max} .**Output:** The trained model parameters Θ .

- 1: In IFEM, X_{ife} is generated by Equations (1), (2) and the set of images I ;
- 2: In LSEM, X_{lsv} and P are generated from L by Equations (3), (4);
- 3: $X^{(1)} = X_{lsv}$, $A = P$;
- 4: $\tilde{A} = (D + I)^{-1}(A + I)$;
- 5: **for** $l = 1 \rightarrow l_{max}$ **do**
- 6: $X^{(l+1)} = \delta\left(\left(\tilde{A} + \text{diag}(\tilde{A})\right) X^{(l)} W^{(l)}\right)$;
- 7: **end for**
- 8: $X_{lse} = X^{l_{max}}$;
- 9: The prediction results \hat{y} are obtained by the dot product of X_{ife} and X_{lse} ;
- 10: Calculate the loss \mathcal{L} according to Equation (9);
- 11: Update Θ with back propagation and the loss \mathcal{L} ;
- 12: **return** Model parameters Θ .

4 Experiments

In this section, first, we present the evaluation metrics for multi-label image recognition. Then, we describe the implementation details of our experiment. And we report the results of our experiments on MS-COCO [13] and VOC 2007 [7]. Finally, we show the results of the ablation experiments, as well as the visualization of the adjacency matrix.

4.1 Evaluation Metrics

For a fair comparison with existing methods, we follow the conventional evaluation metrics as [3, 19]. We adopt the average precision (AP) to reflect the performance of each category and mean average precision (mAP) to reflect the overall performance. Top-3 evaluation metric is also used in our experiments. In addition, we use traditional evaluation metrics in the image classification which are defined in Eq. (10). It includes overall precision, recall, F1-measure (OP, OR, OF1) and average per-class precision, recall, F1-measure (CP, CR, CF1),

$$\begin{aligned}
 \text{OP} &= \frac{\sum_i N_i^{cp}}{\sum_i N_i^p}, & \text{OR} &= \frac{\sum_i N_i^{cp}}{\sum_i N_i^g}, & \text{OF1} &= \frac{2 \times \text{OP} \times \text{OR}}{\text{OP} + \text{OR}}, \\
 \text{CP} &= \frac{1}{C} \sum_i \frac{N_i^{cp}}{N_i^p}, & \text{CR} &= \frac{1}{C} \sum_i \frac{N_i^{cp}}{N_i^g}, & \text{CF1} &= \frac{2 \times \text{CP} \times \text{CR}}{\text{CP} + \text{CR}},
 \end{aligned} \tag{10}$$

where C is the category number of labels. For the i -th label, N_i^{cp} denotes the number of correct predictions among the predicted labels, N_i^p denotes the number of labels that predicted by the model, N_i^g denotes the number of ground truth labels provided by the dataset.

4.2 Implementation Details

In our experiments, the parameter settings we use will be elaborated as follows. The input image is cropped at random centers for data augmentation. Then, each input image is converted to a tensor of dimension $\mathbb{R}^{512 \times 512 \times 3}$ and fed into the CNN. We used a pre-trained RexNeXt-101 to accelerate the model training and added a max pooling to transform the dimension to 2048. In LSEM, unlike the traditional two-layer GCN, we employ a four-layer GCN with input and output dimensions of 300 for the first three layers, and 300 for the fourth layer input, and 2048 for the output dimension. And we adopt GloVe [15] in word-to-vector transformation. We adopt LeakyReLU [12] with the negative slope of 0.18 as the non-linear activation function. Dropout layers with hyperparameter 0.15 are also added to prevent the model from falling into an over-fitting state. In the optimization process, we used a stochastic gradient descent method.

The initial learning rate is set to 0.1. And it changes to one-tenth of the original learning rate every 20 rounds, with a lower limit of 0.001. We train 300 epochs in total and adopt an early stopping strategy to preserve the best results and reduce the impact of over-fitting. We use PyTorch to implement our framework. Our experimental server is configured with Ubuntu 16.04, an Intel Xeon E5-2620 CPU, 128 G of RAM and a Tesla P100 GPU.

4.3 Experimental Results

In order to effectively compare our framework with current state-of-the-art algorithms, we have conducted comparative experiments on mainstream benchmark datasets including MS-COCO and VOC 2007. Figure 3 presents some sample images from both datasets.


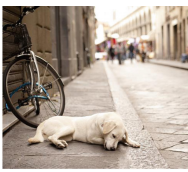


Datasets	Microsoft COCO		PASCAL VOC 2007	
Images				
Ground truths	Person, baseball bat, baseball glove, sports ball	Bicycle, dog, person	Person, horse	Boat, car, person
Prediction results	Person, baseball bat, sports ball	Dog, bicycle, person	Person, horse	Car, person, boat

Fig. 3. Some example images from MS-COCO and VOC 2007 datasets.

Table 1. Comparisons with state-of-the-art methods on the MS-COCO dataset. The best results are marked in bold.

Methods	mAP	Top-3						All					
		CP	CR	CF1	OP	OR	OF1	CP	CR	CF1	OP	OR	OF1
CNN-RNN [19]	61.2	66.0	55.6	60.4	69.2	66.4	67.8	—	—	—	—	—	—
RNN-Attention [20]	—	79.1	58.7	67.4	84.0	63.0	72.0	—	—	—	—	—	—
Order-Free RNN [1]	—	71.6	54.8	62.1	74.2	62.2	67.7	—	—	—	—	—	—
SRN [24]	77.1	85.2	58.8	67.4	87.4	62.5	72.9	81.6	65.4	71.2	82.7	69.9	75.8
Multi-Evidence [8]	—	84.5	62.2	70.6	89.1	64.3	74.7	80.4	70.2	74.9	85.2	72.5	78.4
ML-GCN (Binary) [4]	80.3	84.9	61.3	71.2	88.8	65.2	75.2	81.1	70.1	75.2	83.8	74.2	78.7
MDGCN	81.7	85.0	64.4	73.3	90.2	66.7	76.7	81.5	73.1	77.1	84.7	76.5	80.4

Experimental Results on MS-COCO. Microsoft COCO (MS-COCO) [13] dataset is a large-scale object detection, image segmentation and image description generation dataset. MS-COCO version we used in our experiments is 2014, which contains 82,783 images in the training set and 40,504 images in the validation set, which used as a test set in this experiment. It is divided into 80 classes. It is therefore a challenging dataset for multi-label image recognition tasks.

The results of the comparison experiments on the COCO dataset are shown in Table 1. It can be seen that our algorithm achieves comparable performance and, on average, leads on the performance metrics mAP, CF1 and OF1. Compared to the latest algorithms our method has an advantage in precision, this is because we use a deep graph convolutional network that captures semantic information with higher accuracy.

Experimental Results on VOC 2007. PASCAL Visual Object Classes Challenge (VOC 2007) [7] dataset is now widely used for multi-label image classification tasks. It contains a total of 9963 images, of which 5011 images from the original divided training set plus the validation set are used as the training set in the experiment, and 4952 images from the test set are used as the test set in the experiment. It has 20 classes with an average of 2.9 labels per image.

The results of our experiment are presented in Table 2. The results indicate that our method achieves advanced performance on sixteen of the twenty classes, and comparable performance on the remaining four. The mAP metric reflects the overall accuracy of the model, and in terms of this metric, our model has a performance of 94.8, ahead of ML-GCN’s 94.0. This shows that our model has an advantage in the use of feature extraction for most classes. And in our experiments, the computational time of our method is relatively similar to that of ML-GCN, although we employ a more complex mechanism.

4.4 Ablation Studies

To explore the effectiveness of our MDGCN, we conducted ablation experiments on the picture feature extraction network part. The experimental results are shown in Table 3. It can be seen that the performance of ResNeXt for direct

Table 2. Comparisons of AP and mAP with state-of-the-art methods on the VOC 2007 dataset. The best results are highlighted in bold.

Methods	Aero	Bike	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Table	Dog	Horse	Motor	Person	Plant	Sheep	Sofa	Train	Tv	mAP
VeryDeep [17]	98.9	95.0	96.8	95.4	69.7	90.4	93.5	96.0	74.2	86.6	87.8	96.0	96.3	93.1	97.2	70.0	92.1	80.3	98.1	87.0	89.7
CNN-RNN [19]	96.7	83.1	94.2	92.8	61.2	82.1	89.1	94.2	64.2	83.6	70.0	92.4	91.7	84.2	93.7	59.8	93.2	75.3	99.7	78.6	84.0
FeV+LV [23]	97.9	97.0	96.6	94.6	73.6	93.9	96.5	95.5	73.7	90.3	82.8	95.4	97.7	95.9	98.6	77.6	88.7	78.0	98.3	89.0	90.6
HCP [21]	98.6	97.1	98.0	95.6	75.3	94.7	95.8	97.3	73.1	90.2	80.0	97.3	96.1	94.9	96.3	78.3	94.7	76.2	97.9	91.5	90.9
RNN-Attention [20]	98.6	97.4	96.3	96.2	75.2	92.4	96.5	97.1	76.5	92.0	87.7	96.8	97.5	93.8	98.5	81.6	93.7	82.8	98.6	89.3	91.9
Atten-Reinforce [2]	98.6	97.1	97.1	95.5	75.6	92.8	96.8	97.3	78.3	92.2	87.6	96.9	96.5	93.6	98.5	81.6	93.1	83.2	98.5	89.3	92.0
SSGRL [3]	99.5	97.1	97.6	97.8	82.6	94.8	96.7	98.1	78.0	97.0	85.6	97.8	98.3	96.4	98.8	84.9	96.5	79.8	98.4	92.8	93.4
ML-GCN [4]	99.5	98.5	98.6	98.1	80.8	94.6	97.2	98.2	82.3	95.7	86.4	98.2	98.4	96.7	99.0	84.7	96.7	84.3	98.9	93.7	94.0
MDGCN	99.9	97.4	98.8	98.3	83.5	96.5	98.0	98.7	83.2	97.7	86.9	98.9	98.8	97.1	99.2	85.8	95.8	86.9	99.0	95.7	94.8

multi-label image classification with the same data augmentation technique approach is also higher than that of ResNet. And, our framework, incorporating semantic features, thus outperforms ResNeXt on two datasets.

We tested the effect of DGCN with different network depths. The experimental results in Table 4 show that the best performance is achieved when the depth of DGCN is 4. It also illustrates that although the regularization method has been used to avoid deepening the network with weakened neighbor node connections, a DGCN is not always better.

Table 3. The impact of choosing different CNN on performance.

Datasets	Methods	mAP	CF1	OF1
MS-COCO	ResNet-101 [10]	77.3	72.8	76.8
MS-COCO	RexNeXt-101 [22]	78.2	72.7	76.7
MS-COCO	MDGCN	81.7	77.1	80.4
VOC 2007	ResNet-101 [10]	89.9	83.1	84.5
VOC 2007	RexNeXt-101 [22]	91.6	84.7	86.7
VOC 2007	MDGCN	94.8	88.9	89.9

Table 4. Comparisons with different depths of DGCN.

Depths	MS-COCO			VOC 2007		
	mAP	CF1	OF1	mAP	CF1	OF1
2	81.0	75.9	79.2	94.2	88.0	88.2
3	81.6	76.9	80.3	94.7	88.5	89.7
4	81.7	77.1	80.4	94.8	88.9	89.9
5	81.5	76.6	80.1	94.1	87.8	89.2
6	81.2	76.2	79.3	93.9	87.5	88.1

4.5 Adjacency Matrix Visualization

The adjacency matrix is very important for the label semantic extraction module, therefore we present it visually in Fig. 4. From the visualization results, the original A reflects the co-occurrence probability, from which we can see that the 15-th class of VOC 2007 and the 50-th class of MS-COCO are “person”, and have a high co-occurrence probability with other labels. It is worth noting that compared to the traditional GCN’s A' , we construct an A' with the feature that nearby nodes have more contribution compared to distant nodes.

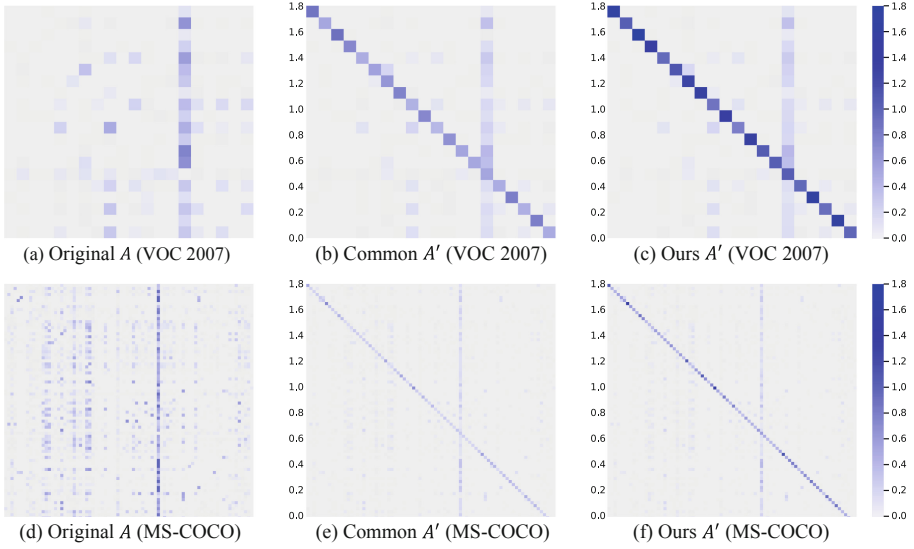


Fig. 4. Visualization of the label co-occurrence matrix A and the matrix A' obtained from it. Original A indicates that we construct it according to Eq. (4), common A' represents that it is constructed according to the original GCN, and ours A' denotes the adjacency matrix constructed by our method.

5 Conclusion

We introduced a new end-to-end multi-label image recognition framework called MDGCN. In order to make full use of the available label information, our framework was designed based on deep graph convolutional networks. In addition, we employed the method of enhancing the diagonal elements of the adjacency matrix to avoid the weakening of its own information during the propagation of DGCN. We conducted validation experiments and we achieved comparable performance. So we believe that our proposed framework can be used for more complex and larger multi-label image recognition tasks.

Acknowledgments. This work is in part supported by the National Natural Science Foundation of China (Grant No. U1705262), the Natural Science Foundation of Fujian Province (Grant Nos. 2020J01130193 and 2018J07005).

References

1. Chen, S., Chen, Y., Yeh, C., Wang, Y.F.: Order-free RNN with visual attention for multi-label classification. In: AAAI, pp. 6714–6721 (2018)
2. Chen, T., Wang, Z., Li, G., Lin, L.: Recurrent attentional reinforcement learning for multi-label image recognition. In: AAAI, pp. 6730–6737 (2018)
3. Chen, T., Xu, M., Hui, X., Wu, H., Lin, L.: Learning semantic-specific graph representation for multi-label image recognition. In: ICCV, pp. 522–531 (2019)

4. Chen, Z., Wei, X., Wang, P., Guo, Y.: Multi-label image recognition with graph convolutional networks. In: CVPR, pp. 5177–5186 (2019)
5. Chiang, W., Liu, X., Si, S., Li, Y., Bengio, S., Hsieh, C.: Cluster-GCN: an efficient algorithm for training deep and large graph convolutional networks. In: ACM SIGKDD, pp. 257–266 (2019)
6. Deng, J., Dong, W., Socher, R., Li, L., Li, K., Li, F.: ImageNet: a large-scale hierarchical image database. In: CVPR, pp. 248–255 (2009)
7. Everingham, M., Gool, L.V., Williams, C.K.I., Winn, J.M., Zisserman, A.: The pascal visual object classes (VOC) challenge. IJCV **88**(2), 303–338 (2010)
8. Ge, W., Yang, S., Yu, Y.: Multi-evidence filtering and fusion for multi-label classification, object detection and semantic segmentation based on weakly supervised learning. In: CVPR, pp. 1277–1286 (2018)
9. Gong, Y., Jia, Y., Leung, T., Toshev, A., Ioffe, S.: Deep convolutional ranking for multilabel image annotation. In: ICLR (2014)
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR, pp. 770–778 (2016)
11. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: ICLR (2017)
12. Maas, A.L., Hannun, A.Y., Ng, A.Y.: Rectifier nonlinearities improve neural network acoustic models. In: ICML, pp. 1–6 (2013)
13. Lin, T.-Y., et al.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
14. Ma, X., et al.: Understanding adversarial attacks on deep learning based medical image analysis systems. Pattern Recogn. **110**, 107332 (2021)
15. Pennington, J., Socher, R., Manning, C.D.: Glove: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, pp. 1532–1543 (2014)
16. Qian, S., Xue, D., Zhang, H., Fang, Q., Xu, C.: Dual adversarial graph neural networks for multi-label cross-modal retrieval. In: AAAI, pp. 2440–2448 (2021)
17. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR (2015)
18. Sun, K., Lin, Z., Zhu, Z.: AdaGCN: adaboosting graph convolutional networks into deep models. In: ICLR (2021)
19. Wang, J., Yang, Y., Mao, J., Huang, Z., Huang, C., Xu, W.: CNN-RNN: a unified framework for multi-label image classification. In: CVPR, pp. 2285–2294 (2016)
20. Wang, Z., Chen, T., Li, G., Xu, R., Lin, L.: Multi-label image recognition by recurrently discovering attentional regions. In: ICCV, pp. 464–472 (2017)
21. Wei, Y., et al.: HCP: a flexible CNN framework for multi-label image classification. IEEE TPAMI **38**(9), 1901–1907 (2016)
22. Xie, S., Girshick, R.B., Dollár, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: CVPR, pp. 5987–5995 (2017)
23. Yang, H., Zhou, J.T., Zhang, Y., Gao, B., Wu, J., Cai, J.: Exploit bounding box annotations for multi-label object recognition. In: CVPR, pp. 280–288 (2016)
24. Zhu, F., Li, H., Ouyang, W., Yu, N., Wang, X.: Learning spatial regularization with image-level supervisions for multi-label image classification. In: CVPR, pp. 2027–2036 (2017)