Recurrent Attention Reinforcement Learning for Multi-label Image Recognition

Anonymous AAAI Submission
Paper ID: 1286

Abstract

Recognizing multiple labels of images is a fundamental but challenging task in computer vision, and remarkable progress has been attained by localizing semantic-aware image regions and predicting their labels with deep convolutional neural networks. The step of hypothesis regions (region proposals) localization in these existing multi-label image recognition pipelines, however, usually takes redundant computation cost, e.g., generating hundreds of meaningless proposals with non-discriminative information and extracting their features, and the spatial contextual dependency modeling among the localized regions are often ignored or over-simplified. To resolve these issues, this paper proposes a recurrent attention reinforcement learning framework to iteratively discover a sequence of attentional and informative regions that are related to different semantic objects and further predict label scores conditioned on these regions. Besides, our method explicitly models long-term dependencies among these attentional regions that help to capture semantic label co-occurrence and thus facilitate multi-label recognition. Extensive experiments and comparisons on two large-scale benchmarks (i.e., PASCAL VOC and MS-COCO) show that our model achieves superior performance over existing state-of-the-art methods in both performance and efficiency as well as explicitly identifying image-level semantic labels to specific object regions.

Introduction

Image classification, as a foundational problem in computer vision, is receiving increasing attention in the research community. Although marked progress is achieved in this topic thanks to the great success of deep convolutional neural networks (CNNs) (Krizhevsky, Sutskever, and Hinton 2012; He et al. 2016), existing approaches mainly focus on single-label image classification that considers the situation where an image would contain only one object. In contrast, multi-label image recognition shows more practical significance, as the real-world image is normally annotated with multiple labels and modeling rich semantic information is essential for the task of high-level image understanding.

A straightforward method that extends CNNs to multi-label image recognition is to fine tune the networks pre-trained on single-label classification dataset (e.g., ImageNet (Russakovsky et al. 2015)) and extract global representation for multi-label recognition (Chatfield et al. 2014). Though being end-to-end trainable, classifiers trained on global image representation may not generalize well to images containing multiple objects with different locations, scales, occlusions, and categories. An alternative way (Yang et al. 2016; Wei et al. 2016) is to introduce object proposals that are assumed to contain all possible foreground objects in the image, and aggregate features extracted from all these proposals to incorporate local information for multi-label image recognition. Despite notable improvement compared to global representation, these methods still have many flaws. First, these methods need to extract hundreds of proposal to achieve a high recall but feeding such a large number of proposals to the CNN for classification is extremely time-consuming. Second, an image usually contains only several objects, most of the proposals either provide intensely coarse information of an object or even refer to the same object. In this way, redundant computation and sub-optimal performance are inevitable, especially in complex scenarios. Last but not least, they usually oversimplify the contextual dependencies among foreground objects and thus fail to capture label correlations in images.

In this paper, inspired by the way that humans continually move fovea from one discriminative object to the next when performing image labeling tasks, we propose an end-to-end trainable recurrent attention reinforcement learning framework to adaptively search the attentional and contextual regions in term of classification. Specifically, our proposed framework consists of a fully convolutional network for extracting deep feature representation, and a recurrent attention-aware module, implemented by an LSTM network, to iteratively locate the class-related regions and predict the label scores over these located regions. At each iteration, it predicts the label scores for the current region and searches an optimal location for the next iteration. Note that by "remember" the information of the previous iterations, the LSTM can naturally capture contextual dependencies among the attentional regions, which is also a key factor that facilitates multi-label recognition (Zhang et al. 2016). During training, we formulate it as a sequential decision-making problem, and introduce reinforcement learning similar to previous visual attention models (Mnih et al. 2014; Ba, Mnih, and Kavukcuoglu 2014), where the action is searching the attentional location of each glimpse and per-

forming classification on attentional regions, the state is the features regarding the current regions as well as the information of previous iteration, and the reward measures the classification correctness. In this way, the proposed framework is trained with merely image-level labels in an end-to-end fashion, requiring no explicit object bounding boxes.

To the best of our knowledge, this is the first paper that introduces recurrent attentional mechanism with deep reinforcement learning into generic multi-label image classification. Compared to the recent hypothesis-regions-based multi-label recognition methods, our proposed method not only enjoys better computational efficiency and higher classification accuracy, but also provides a semantic-aware object discovery mechanism based on merely image-level labels. Experimental results on two large-scale benchmarks (PASCAL VOC and MS-COCO) demonstrate the superiority of our proposed method against state-of-the-art algorithms. We also conduct experiments to extensively evaluate and discuss the contribution of the crucial components.

Related Work

We review the related works according to two main research streams: multi-label image recognition and visual attention networks.

Multi-label image recognition

Recent progress on single-label image classification is made based on the deep convolutional neural networks (CNNs)(Krizhevsky, Sutskever, and Hinton 2012; Simonyan and Zisserman 2014; He et al. 2016) that learn powerful visual representation via stacking multiple nonlinear transformations. Several works have also adapted these single-label classification networks to multi-label image recognition (Sharif Razavian et al. 2014; Simonyan and Zisserman 2014; Yang et al. 2016; Wei et al. 2016; Wang et al. 2016). For example, Razavian et al. (Sharif Razavian et al. 2014) extract off-the-shelf features using Alex-Net pre-trained on the ImageNet dataset and train an SVM classifier for each category. Chatfield et al. (Chatfield et al. 2014) fine-tune the network using the target multi-label dataset to learn more domain-specific features, which helps to boost the classification accuracy. Gong et al. (Gong et al. 2013) explore training the CNN via various losses to tackle this problem and comes to a conclusion that the weighted approximate ranking loss can return the best performance. However, these methods treat the labels in the image independently and fail to capture semantic label co-occurrence. Instead, there are a series of works resorting to graphical models to capture pairwise label correlations, including Conditional Random Field (Ghamrawi and McCallum 2005), Dependency Network (Guo and Gu 2011), and co-occurrence matrix (Xue et al. 2011). Most recently, Wang et al. (Wang et al. 2016) formulate a CNN-RNN framework to jointly characterize the semantic label dependency and the image-label relevance. Zhu et al. (Zhu et al. 2017) further propose a Spatial Regularization Network that generates class-related attention maps and captures both spatial and semantic label dependencies via simple learnable convolutions.

The works mentioned above mainly consider the global representation of the whole image. However, a classifier trained using the global representation may not be optimal for multi-label image recognition, since they not only ignore the relationship between semantic labels and local image regions but also are vulnerable to the non-informative background. To address this problem, recent works (Yang et al. 2016; Wei et al. 2016) extract object proposals as the informative regions and aggregate local features on these regions for multi-label recognition. More concretely, Wei et al. (Wei et al. 2016) propose a Hypotheses-CNN-Pooling framework, which makes predictions on each proposal and then aggregates all the predictions as the final output through category-wise max-pooling. Yang et al. (Yang et al. 2016) formulates the multi-label image recognition as a multi-class multi-instance problem and incorporates feature as well as label view information of the proposals for feature enhancement. Newest work (Zhang et al. 2016) also utilizes CNN-based proposals and simultaneously models label dependencies among the proposals. Despite jointly training the proposals and image classification, this method needs to additionally train the proposal generation component with the annotation of bounding boxes.

Visual attention networks

One drawback of the proposal-based methods is the necessary for extracting object proposals, preventing the model from end-to-end training (Wei et al. 2016; Yang et al. 2016) or requiring extra annotations of the bounding boxes (Zhang et al. 2016). Recently, visual attention networks have been intensively proposed to automatically mine the relevant and informative regions, which have benefited a broad range of vision tasks, including image recognition (Mnih et al. 2014; Ba, Mnih, and Kavukcuoglu 2014; Xiao et al. 2015), image captioning (Xu et al. 2015) and visual question answering (Xiong, Merity, and Socher 2016). These works usually design a recurrent neural network to iteratively search the attentional regions, which can be formulated as a sequential decision-making problem. Reinforcement learning technique is commonly introduced to optimize the sequential model with delayed reward. Specifically, (Mnih et al. 2014; Ba, Mnih, and Kavukcuoglu 2014) propose a recurrent attention model trained with reinforcement learning to attend the most relevant regions of the input image and demonstrate both accurate and efficient results on the digital classification task. However, it may not generalize well to multi-label classification for generic images as they are far more complex, and different objects undergo drastic changes in both scales and shapes. In this paper, we introduce the recurrent attention mechanism into generic multi-label image classification for locating attentional and contextual regions regarding classification and demonstrate it can still improve multi-label recognition in both accuracy and efficiency.

Proposed Method

Figure 1 illustrates an overview of the proposed method. Given an input image I, it is first resized to $W \times H$ and fed into the VGG16 ConvNet (Simonyan and Zisserman 2014).

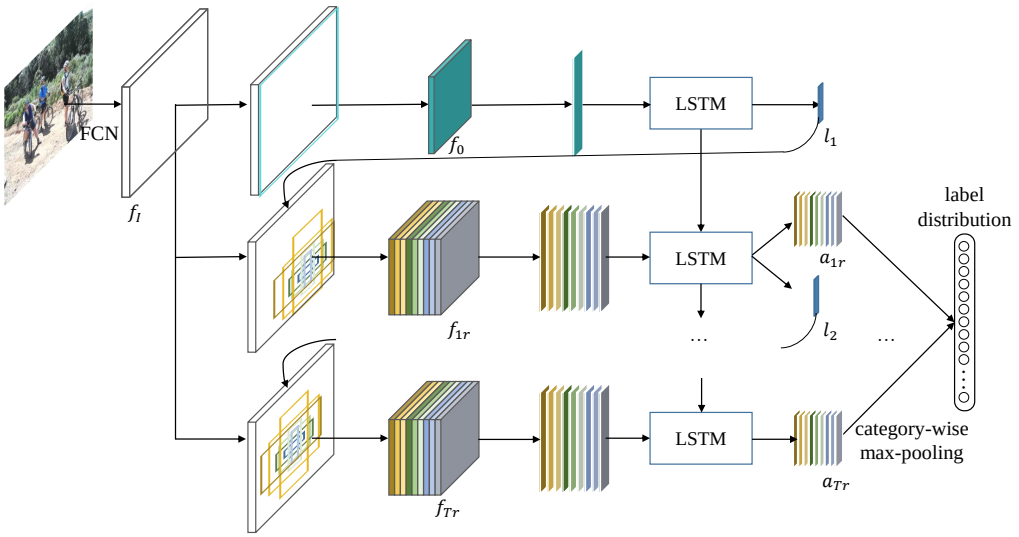

Figure 1: Overview of our proposed framework for multi-label image recognition. The input image is first fed to the VGG16 ConvNet and mapped to the feature maps f_I. At each iteration t, k regions are yielded at the center location l_t estimated from the previous iteration and corresponding fixed-size features are also extracted. An LSTM unit takes these features as well as the hidden state of the previous iteration as input to predict the scores for each region and searches the location for the next iteration. All the predicted scores are fused using the category-wise max-pooling to obtain the final label distribution. The framework is end-to-end trained using merely image-level labels using reinforcement learning techniques.

The ConvNet processes on the whole image with multiple stacked convolutional layers to produce the feature maps $f_I \in \mathcal{R}^{C\times W'\times H'}$. Here, we use the feature maps from the last convolutional layer (i.e., conv5_3). The core of our proposed method is the recurrent attention-aware module that locates the attentional regions and predicts the label scores for these regions in an iterative manner. Finally, the scores over all attentional regions are fused to get the final label distribution. In the following context, we introduce this module in detail.

At iteration t, the agent first receives a location l_t computed at the previous iteration and extracts regions based on l_t. Previous works (Mnih et al. 2014) simply extract square patches centered at l_t. However, general objects undergo drastic changes in both shapes and scales, and thus directly extracting squre patches can hardly cover all these objects. Inspired by the anchor strategy proposed in (Ren et al. 2015), we yield k regions $R_t = \{R_{tr}\}_{r=1}^{k}$ related to various scales and aspect ratios centered at l_t and then extract the features for all these regions:

$$f_{tr} = \mathcal{G}(f_I, R_{tr}), r = 1, 2, \cdots, k, \tag{1}$$

where $\mathcal{G}$ consists of a cropping operation that crops the region R_{tr} on f_I, followed by a bilinear interpolation that maps the cropped feature maps to a fixed-size counterpart f_{tr}. Previous works (Mnih et al. 2014) crop the regions at original input image and apply CNN for repeatedly extracting features for each region, leading to a high computational burden. Instead, we apply the operation on the feature maps f_I to avoid repeating the convolutional processes that are computationally intensive, significantly improving the efficiency during both training and test stages. Once the features are extracted, the recurrent attention-aware module, implemented by an LSTM network (Hochreiter and Schmidhuber 1997), takes the hidden state of the previous iteration as well as the features of currently located regions as input, predicts the classification scores for each region and searches an optimal location for the next iteration, formulated as:

$$\{a_{t1}, a_{t2}, \cdots, a_{tk}, l_{t+1}\} = \mathcal{T}_\pi(f_{t1}, f_{t2}, \cdots, f_{tk}, h_{t-1}; \theta) \tag{2}$$

where $\mathcal{T}_\pi(\cdot)$ represents the recurrent attention-aware module, and θ denotes the network parameters. a_{tr} is the label score vector with respect to the region R_{tr}. The initial region is set as the whole image, so R_0 has only one region, and it is merely used to determine the location l_1.

Category-wise max-pooling. The iterations are repeated for $T+1$ times, yielding $T \times k$ label score vectors, i.e., $\{a_{tr}|t = 1, 2, \cdots, T; r = 1, 2, \cdots, k\}$, where $a_{tr} = \{a_{tr}^0, a_{tr}^1, \cdots, a_{tr}^{C-1}\}$ is the score vector of region R_{tr} over C class labels. Following previous work (Wei et al. 2016), we utilize the category-wise max-pooling operation to fuse these score vectors and obtain the final result $a = \{a^0, a^1, \cdots, a^{C-1}\}$ via simply maximizing out the scores over regions for each category, formulated as:

$$a^c = \max(a_{11}^c, a_{12}^c, \ldots, a_{Tk}^c), c = 0, 1, \ldots, C-1. \tag{3}$$

Recurrent attention-aware module

The recurrent attention-aware module iteratively predicts the label scores of the current regions and searches a most relevant location for the next iteration, which can be regarded as a sequential decision-making problem. At each iteration,

it takes action to predict the label scores for the attended regions and searches an optimal location conditioned on the current states. After the action, the state is updated by a new hidden state and a newly attended location. The process is repeated until a maximum iteration is reached. In the end, the scores of all the located regions are fused to get the final label distribution, and a delayed global reward, which is computed based on this predicted result and the ground-truth labels, is employed to guide the agent training. We elaborate the involved states, actions and reward signal in the following.

State. The state s_t should provide sufficient information for the agent to make decisions. Concretely, it should encode the knowledge of the current environment and those of the previous iterations. To this end, it comprises two parts: 1) the features of the current regions (i.e., $\{f_{tr}\}_{r=1}^{k}$), which is instrumental to classification and provides rich contextual information to help the agent to mine more complemental and discriminative regions; 2) the hidden state of the previous iteration h_{t-1}, which encodes the information of the past iterations and updates over time via the LSTM module. Moreover, simultaneously considering the information of the previous iterations can also help to capture the contextual dependencies among all the glimpsed regions and labels. In this way, by sequentially observing the states $s_t = \{f_{t1}, f_{t2}, \cdots, f_{tk}, h_{t-1}\}$, the agent is capable of performing classification for the current regions and determining the next optimal location.

Action. Given the state s_t, the agent takes two actions: 1) performing classification on the current attentional regions; 2) searching an optimal location l_{t+1} over all possible locations $\{l_{t+1} = (x, y)|0 \leq x \leq W', 0 \leq y \leq H'\}$ on the feature map f_I. As shown in figure 1, a fully-connected layer is utilized to map the extracted features f_{tr} to the semantic representation for each attended region. The LSTM unit takes the semantic representation and the hidden state of the previous iteration as input, and produces a new hidden state h_{tr}. Finally, the classification scores can be computed through a small classification network, denoted as:

$$a_{tr} = f_{cls}(h_{tr}; \theta_{cls}), r = 1, 2, \cdots, k, \tag{4}$$

where the classification network $f_{cls}(\cdot)$ is implemented by a fully-connected layer, with θ_{cls} being its parameters. For the localization action, all the hidden states are first averaged to get a final hidden state, denoted as $h_t = \frac{1}{k}\sum_r h_{tr}$. Then the agent builds a gaussian distribution $P(l_{t+1}|f_{loc}(h_t; \theta_{loc}), \sigma)$. In this equation, $f_{loc}(h_t; \theta_{loc})$, the localization network output, is set as the mean value of the distribution, and σ is its standard deviation and is empirically set as 0.11. Similarly, the localization network $f_{loc}(\cdot)$ is also implemented by a fully-connected layer parameterized by θ_{loc}. At iteration t, the agent selects the localization action l_{t+1} by randomly drawing a location over the probability distribution.

Reward. After executing the actions at each iteration, the agent updates the state and receives a reward signal. For the task of multi-label image recognition, it is desired to aggregate the predictions over all located regions for counting the reward, since each region is expected to be associated with one semantic label. Thus, we define a delayed reward assignment mechanism based on the final aggregated result. For a sample with n ground-truth labels, its label set is $g = \{l_1^g, l_2^g, \cdots, l_n^g\}$. We then sort the predicted scores and obtain the predicted label set $p = \{l_1^p, l_2^p, \cdots, l_n^p\}$ with top-n scores. The reward at iteration t is defined as:

$$r_t = \begin{cases} \frac{|g \cap p|}{n} & t = T \\ 0 & t < T \end{cases} \tag{5}$$

where $|\cdot|$ is the cardinality of the set. We aim to maximize the sum of the discounted rewards:

$$R = \sum_{t=1}^{T} \gamma^{t-1} r_t \tag{6}$$

where γ is the discount factor. We set γ as 1 in our experiments, and the total reward is $R = r_T$. In this work, we utilize the reward to guide the agent to search the optimal actions.

Optimization

At the training stage, in addition to defining the similar classification loss with (Yang et al. 2016), we take the delayed reward assignment into account for optimizing the region localization policy, leading to a hybrid objective function for model training. In the experiments, the model is trained with the hybrid loss in an end-to-end manner.

Formally, the agent needs to learn a policy $\pi((a_t, l_{t+1})|S_t; \theta)$, which predicts a distribution over actions for the current iteration based on the sequence of past observations and actions taken by the agent, i.e., $S_t = R_0, l_1, R_1, a_1, l_2, \cdots, R_t$. To this end, we define the objective function to maximize the expectation of the reward, expressed as:

$$\mathcal{J}(\theta) = \mathbb{E}_{P(S_T;\theta)}[R]. \tag{7}$$

where $P(S_T; \theta)$ is the distribution over all possible interaction sequences, and it is dependent on the policy. Inspired by the work (Mnih et al. 2014), we leverage the REINFORCE algorithm (Williams 1992) from the reinforcement learning community to estimate the gradient for backpropagation. Specifically, it utilizes sample approximation to compute the gradients, formulated as:

$$\begin{aligned} \nabla\mathcal{J}(\theta) &= \sum_{t=1}^{T} \mathbb{E}_{P(S_T;\theta)}[\nabla_\theta \log \pi((a_t, l_{t+1})|S_t; \theta)R] \\ &\approx \frac{1}{M}\sum_{i=1}^{M}\sum_{t=1}^{T}[\nabla_\theta \log \pi((a_t^i, l_{t+1}^i)|S_t^i; \theta)R^i] \end{aligned} \tag{8}$$

where $i = 1, 2, \cdots, M$ denotes the index of the M episodes. However, the gradient estimated using Equation (8) is of high variance, and consequently, the training process is difficult to converge. To solve this problem, we further employ the variance reduction strategy proposed in (Mnih et al. 2014) to obtain an unbiased low-variance gradient estimation.

The policy is learnt using the delayed reward signal, as the "best" action at each iteration is unavailable. In the context of multi-label recognition, the ground-truth labels for each sample exist. Thus, we further define a loss function following (Wei et al. 2016; Yang et al. 2016) as the extra supervision. Suppose there are N training samples, and each sample x_i has its label vector $y_i = \{y_i^0, y_i^1, \ldots, y_i^{C-1}\}$. y_i^c $(c = 0, 1, \ldots, C-1)$ is assigned as 1 if the sample is annotated with the class label c, and 0 otherwise. The ground-truth probability vector of the i-th sample is defined as $\hat{p}_i = y_i / ||y_i||_1$, and the classification loss function is thus formulated as:

$$\mathcal{L}_{\text{cls}} = \frac{1}{N} \sum_{i=1}^{N} \sum_{c=0}^{C-1} (p_i^c - \hat{p}_i^c)^2. \tag{9}$$

where p_i is the predicted probability vector and can be computed via:

$$p_i^c = \frac{\exp(a_i^c)}{\sum_{c'=0}^{C-1} \exp(a_i^{c'})} \; c = 0, 1, \ldots, C-1. \tag{10}$$

Experiments

In this section, we present extensive experimental results and comparisons that demonstrate the superiority of the proposed method. We also conduct experiments to carefully evaluate and discuss the contribution of the crucial components.

Experiment setting

Implementation details During training, all the images are resized to $N \times N$, and randomly cropped with a size of $(N-64) \times (N-64)$, followed by a randomly horizontal flipping, for data augmentation. In our experiments, we train two models with $N = 512$ and $N = 640$, respectively. For the anchor strategy, we set 3 region scales with area 80×80, 160×160, 320×320 for $N = 512$ and 100×100, 200×200, 400×400 for $N = 640$, and 3 aspect ratios of 2:1, 1:1, 1:2 for both scales. Thus, k is set as 9. Both of the models are optimized using the Adam solver with a batch size of 16, an initial learning rate of 0.00001, momentums of 0.9 and 0.999. During testing, we follow (Krizhevsky, Sutskever, and Hinton 2012) to perform ten-view evaluation across the two scales. Specifically, we first resize the input image to $N \times N$ ($N = 512, 640$), and extract five patches (i.e., the four corner patches and the center patch) with a size of $(N-64) \times (N-64)$, as well as their horizontally flipped counterparts. In the experiments, instead of repeatedly extracting features for each patch, we feed the $N \times N$ image to the VGG16 ConvNet and crop the features on the conv5_3 features maps accordingly for each patch. In this way, the computational complexity is remarkably reduced. The model predicts a label score vector for each view, and the final result is computed as the average predictions over the ten views.

Evaluation metrics We first employ the average precision (AP) for each category, and the mean average precision (mAP) over all categories to evaluate all the methods. We also follow (Gong et al. 2013; Wang et al. 2016) to compute the precision and recall for the predicted labels. For each image, we assign top k highest-ranked labels to the image, and compare with the ground-truth labels. The precision is the fraction of the number of the correctly predicted labels in relation to the number of predicted labels; The recall is the fraction of the number of the correctly predicted labels in relation to the number of ground-truth labels. In the experiments, we compute the overall precision, recall, $F1$ (OP, OR, $OF1$) and per-class precision, recall, $F1$ (CP, CR, $CF1$) for comparison, which can be computed as:

$$\begin{aligned} OP &= \frac{\sum_i N_i^c}{\sum_i N_i^p}, & CP &= \frac{1}{C} \sum_i \frac{N_i^c}{N_i^p} \\ OR &= \frac{\sum_i N_i^c}{\sum_i N_i^g}, & CR &= \frac{1}{C} \sum_i \frac{N_i^c}{N_i^g} \\ OF1 &= \frac{2 \times OP \times OR}{OP + OR}, & CF1 &= \frac{2 \times CP \times CR}{CP + CR} \end{aligned} \tag{11}$$

where C is the number of labels, N_i^c is the number of images that are correctly predicted for the i-th label, N_i^p is the number of predicted images for the i-th label, N_i^g is the number of ground truth images for the i-th label.

Comparison with state-of-the-art methods

To prove the effectiveness of the proposed method, we conduct comprehensive experiments on two widely used benchmarks: Pascal VOC 2007 (VOC07) (Everingham et al. 2010) and Microsoft COCO (MS-COCO) (Lin et al. 2014).

Performance on the VOC07 dataset The VOC07 dataset contains 9,963 images of 20 object categories, and it is divided into trainval and test sets. It is the most widely used benchmark for multi-label image recognition, and most competing methods have reported their results on this dataset. We compare our model against the following state-of-the-art methods: FeV+LV (Yang et al. 2016), HCP (Wei et al. 2016), CNN-RNN (Wang et al. 2016), RLSD (Zhang et al. 2016), VeryDeep (Simonyan and Zisserman 2014) and CNN-SVM (Sharif Razavian et al. 2014). Note that we report the results of FeV+LV and HCP using VGG-16 ConvNet for fair comparisons. Following the competitors, we train our model on the trainval set and evaluate the performance on the test set.

The comparison results are summarized in Table 1. As shown, the previous best-performing methods are HCP and FeV+LV, both of which extract hundreds of object proposals, and then aggregate the features of these object proposals for multi-label recognition. They achieve mAPs of 90.9% and 90.6%, respectively. Different from these two methods, our model learns an optimal policy to locate a sequence of discriminative regions, while simultaneously trains classifiers to perform classification on these attended regions. In this way, our model can better explore the relations between semantic labels and attentional regions, leading to the performance improvement. Specifically, our model achieves a mAP of 92.0%, suppressing all the previous state-of-the-art methods by a sizable margin. It is noteworthy that the performance with one single scale of 512 or 640 also performs

Methods	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mAP
CNN-SVM	88.5	81.0	83.5	82.0	42.0	72.5	85.3	81.6	59.9	58.5	66.5	77.8	81.8	78.8	90.2	54.8	71.1	62.6	87.2	71.8	73.9
CNN-RNN	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
VeryDeep	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
RLSD	96.4	92.7	93.8	94.1	71.2	92.5	94.2	95.7	74.3	90.0	74.2	95.4	96.2	92.1	97.9	66.9	93.5	73.7	97.5	87.6	88.5
HCP	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
FeV+LV	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
Ours (512)	98.6	96.9	96.3	94.8	74.1	91.9	96.3	97.1	76.9	91.4	86.2	96.6	96.4	93.1	98.0	79.8	91.7	83.1	98.3	88.6	91.3
Ours (640)	97.9	97.1	96.9	95.3	75.3	91.8	96.5	96.7	76.8	91.0	85.6	95.7	96.0	93.5	98.2	81.0	92.7	80.6	98.2	89.0	91.3
Ours	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

Table 1: Comparison results of AP and mAP in % of our model and the previous state of the art methods on the VOC07 dataset. The best results and second best results are highlighted in red and blue, respectively. Best viewed in color.

better than existing methods, further demonstrating the superiority of our model.

Methods	C-P	C-R	C-F1	O-P	O-R	O-F1
WARP	59.3	52.5	55.7	59.8	61.4	60.7
CNN-RNN	66.0	55.6	60.4	69.2	66.4	67.8
RLSD	67.6	57.2	62.0	70.1	63.4	66.5
Ours (512)	77.5	56.8	65.6	83.0	61.2	70.5
Ours (640)	77.9	56.3	65.4	83.5	61.0	70.5
Ours	78.8	57.2	66.2	84.0	61.6	71.1

Table 2: Comparison results of our model and the previous state of the art methods on the MS-COCO dataset. The best and second best results are highlighted in red and blue, respectively. Best viewed in color.

Performance on the MS-COCO dataset The MS-COCO dataset is originally built for object detection and has also been used for multi-label recognition recently. It is a larger and more challenging dataset, which comprises a training set of 82,081 images and a validation set of 40,137 images from 80 object categories. We compare our model with three state-of-the-art methods, i.e., CNN-RNN (Wang et al. 2016), RLSD (Zhang et al. 2016) and WARP (Gong et al. 2013), on this dataset. Our method and all the competitors are trained on the train set and evaluated on the validation set since the ground truth labels of the test set are unavailable. Following (Wang et al. 2016), when computing the precision recall metrics, we select the top 3 labels for each image. We also filter out the labels with probabilities lower than a predefined threshold (0.1 in our experiments), so the label number of some images would be less than 3.

The comparison results of the overall precision, recall, $F1$, per-class precision, recall, $F1$ are reported in Table 2. Our model significantly outperforms previous methods. Specifically, it achieves a per-class $F1$ score of 66.2%, an overall $F1$ score of 71.1%, beating the previous best method by 4.2% and 3.3%, respectively. Similarly, the performance using single scale is still higher than those of other methods.

Ablation Study

In this subsection, we perform ablative studies to carefully evaluate and discuss the contribution of the critical components of our proposed model.

Effectiveness of the attentional regions The key component of our method is the recurrent attention-aware module that automatically locates the discriminative regions. In this part, we further implement two baseline methods to verify the effectiveness of the attentional regions. The first method replaces the locations attended by our model with randomly selected locations, and it also utilizes 9 anchors for each location. The second method utilizes the representative object proposals as the informative regions to replace the attended regions for classification. This method first employs EdgeBox (Zitnick and Dollár 2014) to extract proposals and adopts non-maximum suppression with a threshold of 0.7 on them based on their objectness scores to exclude the seriously overlapped proposals. The proposals with the top 5 scores are selected. Table 3 presents the comparison results. Our attentional model evidently outperforms these two baseline methods.

Method	mAP (%)
random	89.0
proposal	88.6
attention	90.2

Table 3: Comparison of mAP in % of our model with attentional regions, proposals, and random regions on the VOC07 dataset. The results are evaluated using single-view with the scale of 512×512.

Significance of adopting the LSTM To demonstrate the significance of adopting the LSTM, we have conducted two experiments and reported the results in Table 4. First, we use the LSTM to locate the regions while removing the classification branch and independently classifying the regions by designing a network, obtaining a lower mAP of 90.0%. We further remove the LSTM and also predict the locations independently, obtaining an even lower mAP of 89.6%. These results show the contextual dependencies among the attentional regions captured by the LSTM is crucial for improving the region localization accuracy as well as the multi-label classification accuracy.

Effectiveness of multiple regions with variable scales and aspect ratios As general objects vary dramatically in scale

method	mAP (%)
Ours-A	89.6
Ours-B	90.0
Ours-C	91.3

Table 4: Comparison of mAP in % of using different methods for localization and classification. We report the results using LSTM for both classification and localization (Ours-C), using LSTM for localization but not for classification (Ours-B), and not using LSTM (Ours-A). The results are evaluated using ten-view with the scale of 512×512.

and aspect ratio, we extract 9 regions with 3 scales and 3 aspect ratios, at each iteration. We first visualize some examples of the located regions at each iteration in Figure 2. As shown, different regions can indeed find objects with different scales and aspect ratios. For example, the first image in Figure 2 contains a man and a boat, which vary significantly in scale and aspect ratio. However, both of them can be well located. Concretely, the region with the largest scale and ratio of 2:1 well locates the boat at iteration 4, while the region with the middle scale and ratio of 1:2 can catch the man at iteration 5. The located regions of the second image also exhibit similar results.

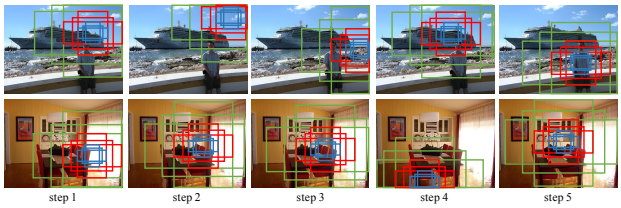

Figure 2: Visualization of the located regions at each iteration. The attentional regions can well locate most semantic objects in the images.

To clearly validate its advantage, we further conduct an experiment that extracts one single region at each location and re-trains the model for comparison. Specifically, we utilize the input image scale of 512×512, and for each predicted location, one region with the size of 224×224 is extracted. The comparison results are depicted in Table 5. It shows that using multiple regions at each iteration leads to better classification performance.

Method	mAP (%)
multiple regions	91.3
single region	90.9

Table 5: Comparison of mAP in % of our model that extract multiple and single regions at each iteration on the VOC07 dataset. The results are evaluated using ten-view with the scale of 512×512.

Analysis of increasing the recursive iteration In this part, we explore the effect of using different recursive iterations T. To this, we train our model with different iterations, i.e., $T = 1, 5, 10$, and report the experimental results in Table 6. When the iteration increases from 1 to 5, the performance has a notable improvement since the located regions may cover more discriminative objects. However, when further increasing the iteration number, the performance does not improve. One possible reason is that when the iteration is greater than 5, the agent has almost mined out all discriminative regions, and locating more regions make litter sense or even bring noise and redundant computation. Thus, in our experiments, the iteration number is set as 5 to better balance the efficiency and effectiveness.

T	mAP (%)
1	90.9
5	91.3
10	91.3

Table 6: Comparison of mAP in % of our model using different recursive iterations on the VOC07 dataset. The results are evaluated using ten-view with the scale of 512×512.

Efficiency analysis

Efficiency is another important metric for the real-world systems. In this part, we analyze the execution time of our model and the previous state-of-the-art methods. We test our model on a desktop with a single NVIDIA GeForce GTX TITAN-X GPU. It takes about 150ms for ten-view evaluation for scale 512 and about 200 ms for scale 640. Thus, the execution time of our method is about 350ms per image. However, recent proposal-based methods, e.g., HCP (Wei et al. 2016) and FeV+LV (Yang et al. 2016), need to compute the proposals and repeat processing hundreds of proposals using the deep CNNs, rendering them extremely inefficient. As shown in (Wei et al. 2016), these methods may take about 10s to process an image on a similar GPU environment, about 30$\times$ slower than ours.

Conclusion

In this paper, we propose a recurrent attention reinforcement learning framework that is capable of automatically locating the attentional and informative regions regarding classification, and predicts the label scores over all attentional regions. We formulate the region localization process as a sequential decision-making problem and resort to reinforcement learning technique to optimize the proposed framework with merely image-level labels in an end-to-end manner. Compared to the previous proposal-based methods, our method can better explore the interaction between semantic labels and attentional regions, while explicitly capturing the contextual dependencies among these regions. Extensive experimental results and evaluations on two large-scale and challenging benchmarks, i.e., Pascal VOC and MicroSoft COCO, well demonstrate the superiority of our proposed method on both accuracy and efficiency.

References

Ba, J.; Mnih, V.; and Kavukcuoglu, K. 2014. Multiple object recognition with visual attention. *arXiv preprint arXiv:1412.7755*.

Chatfield, K.; Simonyan, K.; Vedaldi, A.; and Zisserman, A. 2014. Return of the devil in the details: Delving deep into convolutional nets. *arXiv preprint arXiv:1405.3531*.

Everingham, M.; Van Gool, L.; Williams, C. K.; Winn, J.; and Zisserman, A. 2010. The pascal visual object classes (voc) challenge. *International journal of computer vision* 88(2):303–338.

Ghamrawi, N., and McCallum, A. 2005. Collective multi-label classification. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, 195–200. ACM.

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

Guo, Y., and Gu, S. 2011. Multi-label classification using conditional dependency networks. In *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, 1300.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hochreiter, S., and Schmidhuber, J. 1997. Long short-term memory. *Neural computation* 9(8):1735–1780.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, 1097–1105.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, 740–755. Springer.

Mnih, V.; Heess, N.; Graves, A.; et al. 2014. Recurrent models of visual attention. In *Advances in Neural Information Processing Systems*, 2204–2212.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, 91–99.

Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. 2015. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision* 115(3):211–252.

Sharif Razavian, A.; Azizpour, H.; Sullivan, J.; and Carlsson, S. 2014. Cnn features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 806–813.

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

Wang, J.; Yang, Y.; Mao, J.; Huang, Z.; Huang, C.; and Xu, W. 2016. Cnn-rnn: A unified framework for multi-label image classification. *arXiv preprint arXiv:1604.04573*.

Wei, Y.; Xia, W.; Lin, M.; Huang, J.; Ni, B.; Dong, J.; Zhao, Y.; and Yan, S. 2016. Hcp: A flexible cnn framework for multi-label image classification. *IEEE transactions on pattern analysis and machine intelligence* 38(9):1901–1907.

Williams, R. J. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning* 8(3-4):229–256.

Xiao, T.; Xu, Y.; Yang, K.; Zhang, J.; Peng, Y.; and Zhang, Z. 2015. The application of two-level attention models in deep convolutional neural network for fine-grained image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 842–850.

Xiong, C.; Merity, S.; and Socher, R. 2016. Dynamic memory networks for visual and textual question answering. In *Proceedings of The 33rd International Conference on Machine Learning*, 2397–2406.

Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhutdinov, R.; Zemel, R. S.; and Bengio, Y. 2015. Show, attend and tell: Neural image caption generation with visual attention. *arXiv preprint arXiv:1502.03044*.

Xue, X.; Zhang, W.; Zhang, J.; Wu, B.; Fan, J.; and Lu, Y. 2011. Correlative multi-label multi-instance image annotation. In *2011 International Conference on Computer Vision*, 651–658. IEEE.

Yang, H.; Tianyi Zhou, J.; Zhang, Y.; Gao, B.-B.; Wu, J.; and Cai, J. 2016. Exploit bounding box annotations for multi-label object recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 280–288.

Zhang, J.; Wu, Q.; Shen, C.; Zhang, J.; and Lu, J. 2016. Multi-label image classification with regional latent semantic dependencies. *arXiv preprint arXiv:1612.01082*.

Zhu, F.; Li, H.; Ouyang, W.; Yu, N.; and Wang, X. 2017. Learning spatial regularization with image-level supervisions for multi-label image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Zitnick, C. L., and Dollár, P. 2014. Edge boxes: Locating object proposals from edges. In *European Conference on Computer Vision*, 391–405. Springer.