StackDRL: Stacked Deep Reinforcement Learning for Fine-grained Visual Categorization

Xiangteng He, Yuxin Peng*, and Junjie Zhao
Institute of Computer Science and Technology, Peking University
Beijing 100871, China
pengyuxin@pku.edu.cn

Abstract

Fine-grained visual categorization (FGVC) is the discrimination of similar subcategories, whose main challenge is to localize the quite subtle visual distinctions between similar subcategories. There are two pivotal problems: discovering *which* region is discriminative and representative, and determining *how many* discriminative regions are necessary to achieve the best performance. Existing methods generally solve these two problems relying on the prior knowledge or experimental validation, which extremely restricts the *usability* and *scalability* of FGVC. To address the *"which"* and *"how many"* problems *adaptively* and *intelligently*, this paper proposes a stacked deep reinforcement learning approach (StackDRL). It adopts a *two-stage learning* architecture, which is driven by the *semantic reward* function. *Two-stage learning* localizes the object and its parts in sequence ("which"), and determines the number of discriminative regions adaptively ("how many"), which is quite appealing in FGVC. *Semantic reward* function drives StackDRL to fully learn the discriminative and conceptual visual information, via jointly combining the attention-based reward and category-based reward. Furthermore, *unsupervised discriminative localization* avoids the heavy labor consumption of labeling, and extremely strengthens the *usability* and *scalability* of our StackDRL approach. Comparing with ten state-of-the-art methods on CUB-200-2011 dataset, our StackDRL approach achieves the best categorization accuracy.

1 Introduction

Fine-grained visual categorization (FGVC) is to discriminate the similar subcategories belonging to the same basic category, such as the fine distinction of animals, plants, cars and aircraft models. These subcategories look similarly in the global appearance, and only have a few subtle distinctions in the local parts of the object. It is quite challenging to draw the distinctions between them even for people, not to mention the computer. People prefer to gaze at the object even in the scenario that it is hard to distinguish the object from the background [Neider and Zelinsky, 2006]. Eye movements always tend to direct to the regions of high feature density, texture, and color contrast. These latter influences can all be considered as salient factors affecting object importance. For example, to recognize an image, we always first gaze at where the *object* is, then gaze at those *parts* which are distinct in the object, and finally categorize the image. The two processes of gaze are shown in Figure 1.

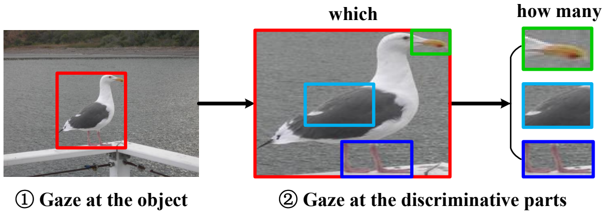

Figure 1: Illustration of the gazes when people recognize an image.

Inspired by the above works, existing FGVC methods focus on the localization of discriminative regions in the image, such as the object and its parts, to achieve good categorization performance. There are two pivotal problems in the discriminative localization: (1) *"Which"* problem: discover which region is more discriminative and representative. (2) *"How many"* problem: determine how many discriminative regions are necessary to achieve the best performance. Existing methods generally solve these problems relying on the prior knowledge or experimental validation, which extremely restricts the usability and scalability of fine-grained visual categorization.

Zhang et al. propose the Part-based R-CNN [Zhang *et al.*, 2014], which utilizes R-CNN [Girshick *et al.*, 2014] with geometric constraints to detect object and its parts. Three discriminative regions are first localized in the image, corresponding to the object, the head and body of the object respectively. Huang et al. propose the Part-Stacked CNN architecture [Huang *et al.*, 2016], which first utilizes a fully convolutional neural network to localize the parts of the object, and then adopts a two-stream network to encode object-level and part-level features simultaneously. Sixteen discriminative

[*] corresponding author

regions are first localized, corresponding to the object and its fifteen annotated parts. We can conclude that the above methods generally address the "*which*" and "*how many*" problems based on the prior knowledge. Researchers rely on the annotated information, such as the ground-truth bounding box and part locations, to determine which region is discriminative and how many discriminative regions are necessary in the categorization. However, not all the annotated information is significant for the categorization. For example, the "eye" part is too similar to draw the distinctions between the similar subcategories, which is not necessary and helpful for categorization. The dependence on prior knowledge makes the discovery process of discriminative regions subjective, and need to customize for different FGVC tasks.

Therefore, most researchers begin to avoid relying on the prior knowledge, and focus on automatically discovering which region is discriminative. Xiao et al. propose a two-level attention method [Xiao *et al.*, 2015], which utilizes the attention mechanism of the convolutional neural networks (CNNs) to select region proposals corresponding to the object and its parts, avoiding the utilization of the ground-truth bounding box and part locations. In this method, two discriminative regions are used. Zhang et al. incorporate deep convolutional filters for both parts selection and description [Zhang *et al.*, 2016]. In this method, the number of discriminative regions is set to 6 in order to achieve the best categorization accuracy. These methods localize the discriminative regions through weakly supervised localization method, but generally set the number of discriminative regions based on the experimental validation. For example, Zhang et al. set the number of discriminative regions as 6 for CUB-200-2011 dataset [Wah *et al.*, 2011], but 5 for Stanford Dogs dataset [Khosla *et al.*, 2011] in their experiments [Zhang *et al.*, 2016]. Besides, the number of discriminative regions is set fixed for each subcategory, which is contrary to the fact that the numbers of representative characteristics are different in variant subcategories. The experimental strategy of setting how many discriminative regions exploited and the fixed number setting are not flexible and limited for fine-grained visual categorization.

Therefore, for addressing the "*which*" and "*how many*" problems *adaptively* and *intelligently*, this paper proposes a stacked deep reinforcement learning approach (StackDRL). It adopts a *two-stage learning* architecture, and its learning is driven by *semantic reward* function. The main contributions of our StackDRL approach can be summarized as follows:

(I) Two-stage learning is proposed to localize the object and its parts ("which") in a sequential way, and determines the number of discriminative regions ("how many") adaptively, which is quite appealing in fine-grained visual categorization. The Stage-I DRL, named *ObjectDRL*, is to remove the background noise in object alignment, only reserve the foreground. The Stage-II DRL, named *PartDRL*, is to further mine the compelling regions of the object, which are variant in numbers and scales for different subcategories. They provide different but complementary visual information to boost the fine-grained representation learning as well as the categorization accuracy.

(II) Semantic reward function is proposed to drive StackDRL to fully learn the valuable and characteristic visual information, via jointly combining the *attention-based reward* and *category-based reward* functions. Attention-based reward improves the localization accuracy by driving StackDRL to localize the regions with more discriminative information. Category-based reward improves the localization accuracy by driving StackDRL to localize the regions with more conceptual information. Both of them boost the performance of fine-grained localization and categorization.

(III) Unsupervised discriminative localization is to further explore the localization performance in an unsupervised manner without using any annotated information, such as category label, ground-truth bounding box and part locations. It avoids the heavy labor consumption of labeling, and extremely strengthens the *usability* and *scalability* of our StackDRL approach, boosting the practical application of fine-grained visual categorization.

2 Stacked Deep Reinforcement Learning

In this section, we detail the proposed stacked deep reinforcement learning approach (StackDRL). It decomposes the discriminative localization learning into two stages, as shown in Figure 2. **(I) Stage-I DRL (ObjectDRL)**: It distinguishes the object from the background, and represents the features of the global appearance. **(II) Stage-II DRL (PartDRL)**: It discovers the characteristics of the object, and tries to draw the distinctions between similar subcategories.

2.1 Problem Formulation

For a given image I, we formulate the discriminative localization as the problem of maximizing a confidence function $f_c : B \rightarrow \mathbb{B}$ over the set of image regions B:

$$b^* = arg \max_{b \in B} f_c(b) \tag{1}$$

We solve the problem via Markov decision process (MDP), which is well suitable for modeling the discrete time sequential decision making process. In the MDP, there define a set of actions A, a set of states S, and a reward function R. In the following subsections, we introduce the details of ObjectDRL and PartDRL respectively, as well as the training details of our StackDRL model.

2.2 ObjectDRL

In ObjectDRL, the given image I is considered as the environment, in which the agent localizes a region at each step by conducting one action of A. The agent has the corresponding state after conducting one action, which contains the information of the current localized region and the action history. For each action, a corresponding reward, may be positive or negative, will feedback to the agent at the training phase. The goal of the agent is to localize a region that accurately contains the target object.

Discriminative Localization Actions

Inspired by Tree-RL [Jie *et al.*, 2016], we define the set of discriminative localization actions A as two action groups due to their different effects, as shown in Figure 3. ObjectDRL and PartDRL adopt the different action groups. In ObjectDRL, only action group 1 is adopted. It consists of five scaling

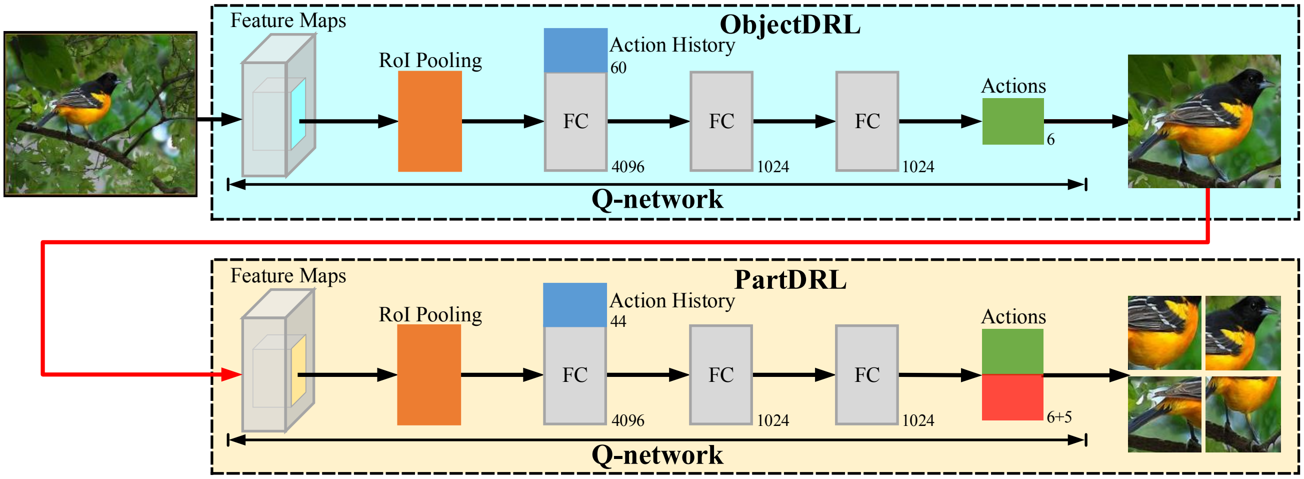

Figure 2: An overview of the proposed StackDRL approach.

actions that can be applied to localize the region of object, and one action to terminate the localization process. Each of the scaling actions scales the current region to a certain sub-region with the scale ratio α, corresponding to scaling the current region to the top left corner, bottom left corner, top right corner, bottom right corner and the center, where $\alpha \in [0, 1]$, set to 0.9 in our experiments. The scaling actions can localize the objects with different scales, which boosts the scalability of the localization. Once an action is conducted on the current region, the content of the region will be changed deterministically, which means that the state is changed. Finally, the only action that does not scale the current region is a trigger to indicate that the current region accurately contains the target object.

States

At action step t, the state of the agent is represented as $S_t = (v_t, h_t)$, where v_t denotes the feature vector of the current localized region in the image, h_t denotes the history vector of conducted actions. The following paragraphs introduce the details of v_t and h_t.

The feature vector $v_t \in \mathbb{R}^d$ is the output of one layer in the CNN model, which is pre-trained on the ImageNet dataset [Deng *et al.*, 2009]. In our experiment, we apply the 16-layer VGGNet [Simonyan and Zisserman, 2014] as the basic CNN model. Feature maps of the layer "con5_3" are extracted as the initial features, and then are connected to a fully-connected layer with 4096 neurons. So the dimension of the final feature vector $d = 4096$. Inspired by Fast R-CNN [Girshick, 2015], RoI Pooling is applied to accelerate feature extraction.

The history vector $h_t \in \mathbb{R}^{|A_O| \times N_{step}}$ is a binary vector to denote which actions have been conducted in the past. Each action is represented by a $|A_O|$-dimensional binary vector, where only the action has been conducted, its corresponding value is 1, otherwise 0. No more than one value is 1 due to the fact only one action is conduct at each action step. $|A_O| = 6$ in ObjectDRL. N_{step} denotes a pre-defined maximal action execution number per image, set to 10 in our experiments, which makes a trade-off between localization speed and accuracy.

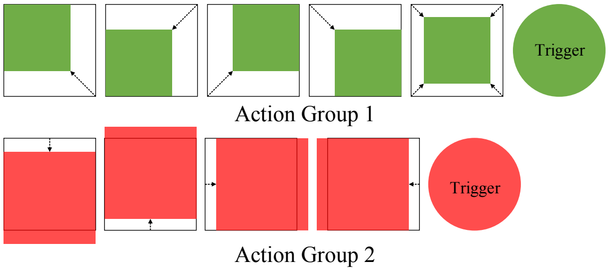

Figure 3: An overview of the discriminative localization actions.

Semantic Reward Function

The reward function R reflects the effect of the conducted action to the localization accuracy, where a positive reward means the conducted action is a good decision to make the localization more accurate, vice versa. We propose a semantic reward function to fully learn the discriminative and conceptual visual information, which consists of attention-based and category-based reward functions.

(I) Attention-based Reward Function

Intersection-over-Union (IoU) between the current region and the ground-truth bounding box of target object is widely used to measure the effect of the conducted action [Jie *et al.*, 2016]. The reward function $RA_a(s, s')$ denotes the reward received when the agent changes from state s to state s' by conducting action a, and its definition is as follows:

$$RA_a(s, s') = sign(IoU(b', g) - IoU(b, g)) \qquad (2)$$

where b denotes the current region, b' denotes the region obtained by conducting action a on the current region b, g denotes the ground-truth bounding box of target object, and $IoU(b, g) = area(b \cap g)/area(b \cup g)$, so is $IoU(b', g)$.

It relies on the ground-truth bounding box, whose labeling is expensive. Therefore, we propose a new reward function based on the attention information, which avoids the heavy labor consumption. Inspired by CAM [Zhou *et al.*, 2016], we first extract the attention map M of the image, which indicates the representative regions used by the CNN to identify the subcategory of image.

Given an image I, the activation of neuron u in the last convolutional layer at spatial location (x, y) is defined as $f_u(x, y)$. The saliency value at spatial location (x, y) is computed as follows:

$$M(x, y) = \frac{1}{|u|} \sum_u f_u(x, y) \tag{3}$$

where $M(x, y)$ directly indicates the importance of activation at spatial location (x, y) for categorizing the image. We perform binarization operation on the saliency map with OTSU algorithm [Otsu, 1979], and take the bounding box that covers the largest connected area as g_{atten}. Therefore, the attention-based reward function is defined as follows:

$$RA_a(s, s') = sign(IoU(b', g_{atten}) - IoU(b, g_{atten})) \tag{4}$$

The attention-based reward function fully utilizes the saliency information of the image, and guides the agent to localize the region with the highest saliency, which is corresponding the region of the target object.

(II) Category-based Reward Function

As is known to all, the category label directly provides the conceptual information. It can guide the agent to localize the region that is actually helpful for the categorization. Therefore, we propose the category-based reward function, which is defined as follows:

$$RC_a(s, s') = sign(P_c(b') - P_c(b)) \tag{5}$$

where function $P_c(\bullet)$ indicates the predicted score of the corresponding region is categorized as subcategory c, c is the annotated category label. The semantic reward function RO in ObjectDRL jointly considers the attention and category information, and its definition is as follows:

$$RO_a(s, s') = RA_a(s, s') + RC_a(s, s') \tag{6}$$

It is noted that we define a different reward function for the trigger following AOL [Caicedo and Lazebnik, 2015]. Its definition is as follows:

$$RO_{trigger}(s, s') = \begin{cases} +\eta, & if\ IoU(b, g_{atten}) \geq \tau \\ -\eta, & otherwise \end{cases} \tag{7}$$

where η is the trigger reward, and the trigger will be conducted when the IoU value is over the threshold τ. η and τ are set to 3 and 0.5 respectively.

2.3 PartDRL

Through the ObjectDRL, the region of the object is obtained. In the PartDRL, we further learn to discover finer and more discriminative regions on the localized object region, which can represent the object with more localized and discriminative information. Like ObjectDRL, it has the corresponding actions, states and reward function.

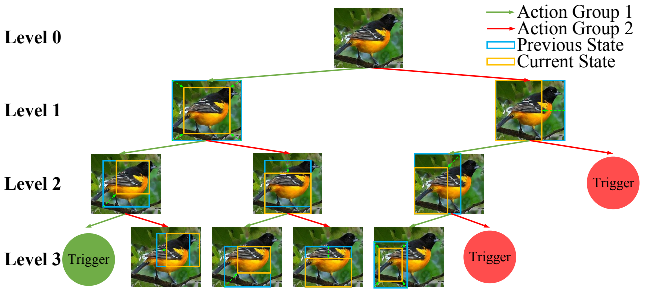

Figure 4: Illustration of the tree structure scheme.

Discriminative Localization Actions

In PartDRL, we hope the agent can localize multiple regions with different characteristics that contribute to discriminating the similar subcategories. Unlike ObjectDRL, we adopt both action group 1 and 2, as shown in Figure 3. Action group 1 has been described in Section 2.2. Action group 2 consists of four local translation actions that can be applied to localize different regions of the object, and also one action to terminate the localization process. Each of the local translation actions moves the region downward, upward, towards the right, towards the left respectively by β times of the current region size, where $\beta \in [0, 1]$, set to 0.1 in our experiments. In order to localize multiple different regions, we follow Tree-RL [Jie *et al.*, 2016] to adopt a tree-structured search scheme, which has two branches, one only conducts actions in group 1, and the other one only conducts actions in group 2, as shown in Figure 4.

States

The state is similar with ObjectDRL except for the history vector. In PartDRL, the history vector is represented as $h_t \in \mathbb{R}^{|A_P| \times N_{level}}$, where $|A_P| = 11$, N_{level} denotes a pre-defined maximal level of the tree structure, set to 4 to balance the localization speed and accuracy.

Semantic Reward Function

The semantic reward function RP in PartDRL also consists of attention-based and category-based reward functions, and its definition is as follows:

$$RP_a(s, s') = RA_a(s, s') + RC_a(s, s') \tag{8}$$

where RC is the same as ObjectDRL, RA is defined as follows:

$$RA_a(s, s') = sign(Mean(b') - Mean(b)) \tag{9}$$

where function $Mean(\bullet)$ denotes the mean value of the attention map of the current region. Through the tree-structured search scheme and the attention-based reward function, we can localize different regions of the object, which can boost the variance of the feature representation. Besides, the reward function of trigger is same as ObjectDRL.

2.4 Q-learning for Discriminative Localization

We apply reinforcement learning to learn the discriminative policy of maximizing the sum of the received rewards of running an episode starting from the original image. Deep

Q-network [Mnih *et al.*, 2015] is applied to solve the reinforcement learning problem. The detailed architecture of our Q-network is shown in Figure 2. Unlike [Jie *et al.*, 2016; Caicedo and Lazebnik, 2015], we use the fine-tuned CNN as the feature extractor of the localized regions at each action step. The CNN is first pre-trained on ImageNet dataset, and then fine-tuned on the specific fine-grained dataset, such as CUB-200-2011 [Wah *et al.*, 2011]. It is because that fine-tuned CNN can obtain a better attention map for each image, and extract more powerful and discriminative features. At training phase, the parameters of Q-network are updated by the agent running multiple episodes, whose behavior is ϵ-greedy [Sutton and Barto, 1998]. In each step, the agent randomly selects an action from the whole action set with probability ϵ, and selects the best action in action group 1 for ObjectDRL with probability $1 - \epsilon$, a random action from the two best actions in action group 1 and 2 for PartDRL with probability $1 - \epsilon$. The update process of Q-network follows Tree-RL [Jie *et al.*, 2016].

2.5 Unsupervised Discriminative Localization

In this subsection, we explore to localize the discriminative regions in an unsupervised manner, which means none of the annotations is used in the localization process. From Section 2.2, we know that attention map can tell which region is significant for categorization. Besides, we know that CNN pre-trained on ImageNet dataset has a good generalization. Considering the attention map extracted from pre-trained CNN has bad ability to reflect the region of object but correspond to some discriminative local regions, we only explore the PartDRL in the unsupervised manner. Specifically, in unsupervised discriminative localization, localization actions and states are the same as PartDRL, which are described in Section 2.3. To avoid using the annotations, we design the semantic reward function RU with attention-based reward, and its definition is the same as RA in PartDRL. But the CNN, which is used to extract the attention map for each image, while is not fine-tuned on the specific fine-grained visual categorization dataset. It is a pre-trained CNN on the ImageNet dataset, which is widely used in the computer vision tasks. So fine-grained subcategory label is not used. Besides, for the Q-learning in discriminative localization, we initialize the parameters of convolutional layers and the first fully-connected layer with the pre-trained CNN, and initialize the parameters of the other fully-connected layers from a zero-mean normal distribution with standard deviation 0.01. Its training process is the same as PartDRL as described in Section 2.3, which is guided by the semantic reward function RU. Thus, there is no annotation used in the whole learning process.

2.6 Final Prediction

For a given image I, no more than $N_{step} - 1$ regions are obtained that correspond to the target object in ObjectDRL, and no more than $2^{N_{level}} - 2$ regions are obtained that correspond to the discriminative regions of the object in PartDRL. Each region is fed to the fine-tuned CNN, i.e. 19-layer VGGNet with batch normalization [Ioffe and Szegedy, 2015], and received its prediction vector v_{score}. For the regions obtained by ObjectDRL, we select the region with highest predicted score. For the regions obtained by PartDRL, we select the region with highest predicted score for each level of the tree structure, and average the scores of the regions to get the predicted score of PartDRL. Finally, the final prediction is obtained by fusing the above predictions.

Table 1: Comparisons with state-of-the-art methods on CUB-200-2011 dataset.

Methods	Train Anno.	Test Anno.	Acc. (%)
Our StackDRL Approach			**86.61**
CVL [He and Peng, 2017a]	text		85.55
RA-CNN [Fu *et al.*, 2017]			85.30
HCA [Cai *et al.*, 2017]			85.30
PNA [Zhang *et al.*, 2017]			84.70
TSC [He and Peng, 2017b]			84.69
LRBP [Kong and Fowlkes, 2017]			84.21
STN [Jaderberg *et al.*, 2015]			84.10
NAC [Simon and Rodner, 2015]			81.01
Coarse-to-Fine [Yao *et al.*, 2016]	bbox		82.50
Coarse-to-Fine [Yao *et al.*, 2016]	bbox	bbox	82.90
PG Alignment [Krause *et al.*, 2015]	bbox	bbox	82.80

3 Experiments

In this section, we present comprehensive experimental results and analyses of our StackDRL approach on CUB-200-2011 dataset [Wah *et al.*, 2011], and adopt Top-1 accuracy to evaluate its effectiveness.

3.1 Implementation Details

We describe the details of StackDRL in the following four aspect: (1) For actions, the ratios of scaling action and local translation actions are set to 0.9 and 0.1 respectively. The maximal action execution number N_{step} is set to 10, and the level of tree structure N_{level} is set to 4. (2) For reward function, the trigger reward η and threshold τ are set to 3 and 0.5 respectively. (3) For Q-learning, the architecture of Q-network is shown in Fig. 2. The region features are computed via RoI Pooling layer with the size of $512 \times 7 \times 7$, and then concatenated with the action history vector to feed into fully-connection layers. Finally, mean squared error (MSE) is used to estimate the predicted values of the localization actions. We initialize the parameters of convolutional layers and the first fully-connected layer with the fine-tuned CNN, and initialize the parameters of the other fully-connected layers from a zero-mean normal distribution with a standard deviation 0.01. In the training phase, the parameter ϵ starts with 1.0 and decreases by 0.1 for each epoch. It is finally fixed to 0.1 after the first 10 epochs to make the agent focus on learning from experiences generated by its own model. Thus we obtain the model of StackDRL.

3.2 Comparisons with State-of-the-art Methods

This subsection presents the experimental results and analyses of our StackDRL approach comparing with ten state-of-the-art methods on CUB-200-2011 dataset, as shown in Table 1. For fair comparison, the annotations of training and testing phases are listed, where “bbox” means the ground-truth bounding box of the object in the image, and “text” denotes the textual descriptions of the image. If the column is empty, it means no annotation is used.

Our approach achieves the best categorization accuracy among all the methods under the same setting that neither the ground-truth bounding box nor part locations. The best result is achieved by CVL [He and Peng, 2017a], which jointly models visual and textual information. It utilizes extern textural descriptions of the image in the training phase. But our StackDRL approach still achieves a 1.06% improvement than CVL. RA-CNN [Fu *et al.*, 2017] achieves the categorization accuracy of 85.30%, which utilizes three regions in different scales. Our StackDRL approach is 1.31% higher than it. PNA trains eleven part detectors to localize the discriminative regions. TSC [He and Peng, 2017b] localizes three discriminative regions to achieve the better categorization accuracy, including one object and two discriminative parts. These existing methods generally set the number of discriminative regions depended on the prior knowledge or experimental validation, which is not flexible and limited for fine-grained visual categorization. While our StackDRL ties to solve this problem, via adaptively localizing the discriminative regions for different images in different subcategories, and achieves the best categorization accuracy. Even comparing with the methods which utilize the ground-truth bounding box in training phase or even in testing phase, our StackDRL approach still achieves better categorization accuracy.

3.3 Effectiveness of Each Component in StackDRL

Detailed experiments are performed on our StackDRL approach in the following aspects:

Effectiveness of Each Stage in StackDRL

From Table 2, we can observe that:

(I) Comparing with the "Baseline", which utilizes the fine-tuned 19-layer VGGNet to recognize the original image, Our PartDRL brings a 2.41% (80.82% → 83.23%) improvement. It is because the good ability of PartDRL to localize the discriminative regions. These regions point out the subtle and local distinctions that are distinguished from other similar subcategories. PartDRL enhances the feature representation with more variances and discrimination.

(II) ObjectDRL boosts the categorization accuracy significantly, which brings a 4.47% improvement compared with "Baseline". It is also 2.06% higher than PartDRL. It is because that the localized region of ObjectDRL contains both the global features reflecting the appearance, and the local features reflecting the salient visual information.

(III) The combination of ObjectDRL and PartDRL can further achieve more accurate result than only one-stage DRL, i.e. 86.61% vs. 85.29% and 83.23%. Comparing with "Baseline", an improvement of 5.79% is achieved. It shows the complementarity of ObjectDRL and PartDRL as well as the effectiveness of the two-stage learning architecture. ObjectDRL and PartDRL have different but complementary gazes at different regions of the image, providing more salient and variant visual information to boost the fine-grained representation learning as well as the categorization.

Effectiveness of Semantic Reward Function

We conduct experiments to show the effectiveness of the proposed semantic reward function. "RA" denotes the attention-based reward functions, and "RC" denotes the category-based reward function. From Table 3, we can observe that: (I) Attention-based reward and category-based reward achieve similar categorization accuracy, which shows that the attention information and category information play similar roles in the fine-grained visual categorization. (II) The joint application of attention-based and category-based reward functions further improve the categorization accuracy due to the fact that the two reward functions focus on different but complementary aspects: attention-based reward provides the discriminative visual information, and category-based reward provides the conceptual visual information.

Table 2: Effectiveness of each stage in StackDRL.

Methods	Acc. (%)
Our StackDRL Approach	**86.61**
ObjectDRL	85.29
PartDRL	83.23
Baseline	80.82

Table 3: Effectiveness of semantic reward function.

Methods	Acc. (%)
Our StackDRL Approach	**86.61**
RA	85.79
RC	85.23

Table 4: Effectiveness of UDL.

Methods	Acc. (%)
Our StackDRL Approach	**86.61**
UDL	83.29
PartDRL	83.23

3.4 Effectiveness of UDL

In this subsection, we explore the effectiveness of unsupervised discriminative localization (denoted as "UDL" in Table 4) in fine-grained visual categorization task. From Table 4, we can see that the application of UDL achieves a promising performance. It is an interesting and significant phenomenon that UDL achieves the similar categorization accuracy with PartDRL, while PartDRL utilizes the category label information. This is own to the good generation of CNN model trained on ImageNet dataset. UDL even outperforms the methods using the ground-truth bounding box, such as Coarse-to-Fine (82.50% and 82.90%) [Yao *et al.*, 2016] and PG Alignment (82.80%) [Krause *et al.*, 2015] shown in Table 1. This inspires us to further explore the study and application of unsupervised discriminative localization.

4 Conclusion

This paper proposes the StackDRL for fine-grained visual categorization. StackDRL localizes the object and its parts via the two-stage learning process automatically, and determines the number of discriminative regions adaptively. Its learning optimization process is driven by semantic reward function, which leads StackDRL to fully learn the discriminative and conceptual visual information, so that the performance of the localization and categorization is improved at the same time. Furthermore, in this paper, we also explore the performance of unsupervised discriminative localization, which is verified to achieve promising performance. It strengthens the usability and scalability of our StackDRL approach. Comparing with ten state-of-the-art methods on FGVC dataset, our StackDRL approach achieves the best performance.

In the future, we devote to improving this work in the following two aspects: First, unsupervised discriminative localization achieves promising results, but is only applied in Part-DRL. We will further explore to apply it in ObjectDRL and bring more improvement in categorization accuracy. Second, in the experimental process, we find the training phase of DRL is time consuming and has instability. We will focus on how to train DRL faster and more stable to achieve better performance. Both of these two aspects will be employed to further improve the FGVC performance.

Acknowledgments

This work was supported by National Natural Science Foundation of China under Grant 61771025 and Grant 61532005.

References

[Cai *et al.*, 2017] Sijia Cai, Wangmeng Zuo, and Lei Zhang. Higher-order integration of hierarchical convolutional activations for fine-grained visual categorization. In *CVPR*, pages 511–520, 2017.

[Caicedo and Lazebnik, 2015] Juan C Caicedo and Svetlana Lazebnik. Active object localization with deep reinforcement learning. In *ICCV*, pages 2488–2496. IEEE, 2015.

[Deng *et al.*, 2009] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. pages 248–255, 2009.

[Fu *et al.*, 2017] Jianlong Fu, Heliang Zheng, and Tao Mei. Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition. In *The CVPR*, July 2017.

[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.

[Girshick, 2015] Ross Girshick. Fast r-cnn. In *ICCV*, December 2015.

[He and Peng, 2017a] Xiangteng He and Yuxin Peng. Fine-grained image classification via combining vision and language. In *The CVPR*, July 2017.

[He and Peng, 2017b] Xiangteng He and Yuxin Peng. Weakly supervised learning of part selection model with spatial constraints for fine-grained image classification. In *AAAI*, pages 4075–4081, 2017.

[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. *arxiv:1502.03167*, 2015.

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

[Jie *et al.*, 2016] Zequn Jie, Xiaodan Liang, Jiashi Feng, Xiaojie Jin, Wen Lu, and Shuicheng Yan. Tree-structured reinforcement learning for sequential object localization. In *NIPS*, pages 127–135, 2016.

[Khosla *et al.*, 2011] Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao, and Fei-Fei Li. Novel dataset for fine-grained image categorization: Stanford dogs. In *CVPR*, volume 2, 2011.

[Kong and Fowlkes, 2017] Shu Kong and Charless Fowlkes. Low-rank bilinear pooling for fine-grained classification. In *CVPR*, pages 7025–7034. IEEE, 2017.

[Krause *et al.*, 2015] Jonathan Krause, Hailin Jin, Jianchao Yang, and Li Fei-Fei. Fine-grained recognition without part annotations. In *CVPR*, pages 5546–5555, 2015.

[Mnih *et al.*, 2015] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.

[Neider and Zelinsky, 2006] Mark B Neider and Gregory J Zelinsky. Searching for camouflaged targets: Effects of target-background similarity on visual search. *Vision research*, 46(14):2217–2235, 2006.

[Otsu, 1979] Nobuyuki Otsu. A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1):62–66, 1979.

[Simon and Rodner, 2015] Marcel Simon and Erik Rodner. Neural activation constellations: Unsupervised part model discovery with convolutional networks. In *ICCV*, pages 1143–1151, 2015.

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

[Sutton and Barto, 1998] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

[Wah *et al.*, 2011] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[Xiao *et al.*, 2015] Tianjun Xiao, Yichong Xu, Kuiyuan Yang, Jiaxing Zhang, Yuxin Peng, and Zheng Zhang. The application of two-level attention models in deep convolutional neural network for fine-grained image classification. In *CVPR*, pages 842–850, 2015.

[Yao *et al.*, 2016] Hantao Yao, Shiliang Zhang, Yongdong Zhang, Jintao Li, and Qi Tian. Coarse-to-fine description for fine-grained visual categorization. *IEEE TIP*, 25(10):4858–4872, 2016.

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

[Zhang *et al.*, 2016] 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.

[Zhang *et al.*, 2017] Xiaopeng Zhang, Hongkai Xiong, Wengang Zhou, Weiyao Lin, and Qi Tian. Picking neural activations for fine-grained recognition. *IEEE TMM*, 19(12):2736–2750, 2017.

[Zhou *et al.*, 2016] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, 2016.