Rethink, Revisit, Revise: A Spiral Reinforced Self-Revised Network for Zero-Shot Learning

Zhe Liu, Yun Li, Lina Yao, *Senior Member, IEEE*, Julian McAuley, and Sam Dixon

***Abstract*— Current approaches to zero-shot learning (ZSL) struggle to learn generalizable semantic knowledge capable of capturing complex correlations. Inspired by *Spiral Curriculum*, which enhances learning processes by revisiting knowledge, we propose a form of spiral learning that revisits visual representations based on a sequence of attribute groups (e.g., a combined group of *color* and *shape*). Spiral learning aims to learn generalized local correlations, enabling models to gradually enhance global learning and, thus, understand complex correlations. Our implementation is based on a two-stage reinforced self-revised (RSR) framework: *preview* and *review*. RSR first previews visual information to construct diverse attribute groups in a weakly supervised manner. Then, it spirally learns refined localities based on attribute groups and uses localities to revise global semantic correlations. Our framework outperforms state-of-the-art algorithms on four benchmark datasets in both zero-shot and generalized zero-shot settings, which demonstrates the effectiveness of spiral learning in learning generalizable and complex correlations. We also conduct extensive analysis to show that attribute groups and reinforced decision processes can capture complementary semantic information to improve predictions and aid explainability.**

***Index Terms*— Neural network, reinforcement learning, spiral learning, zero-shot learning (ZSL).**

I. Introduction

ZERO-SHOT learning (ZSL) aims to learn general semantic correlations between visual attributes (e.g., *is black* and *has tail*) and classes [1]. The correlations allow knowledge transfer from seen to unseen classes and, thus, enable ZSL to classify unseen classes based on the shared knowledge [2]–[5]. Recent ZSL methods have paved the way by adopting extractors or attention for localized attribute knowledge to enhance semantic learning [6]–[10], but localities may be biased toward high-frequency visual features of seen classes. For example, localities, focusing on capturing highly relevant attributes to classes, may build a biased "short-cut" of individual attributes toward seen classes (e.g., *wings* to *Bat*) to maximize the training likelihood. The individual correlations will confuse the prediction of unseen classes with similar features (e.g., *Bird*).

To ease biased visual learning, recent ZSL efforts have achieved success in regularizing the learned correlations with attribute groups, i.e., grouping attributes into high-level semantic groups [4], [11]–[14]. For example, Xu *et al.* [4] refine localities with predefined attribute groups (e.g., *hairless* and *furry* belonging to *texture*). The attribute groups can generalize localities from individual correlations to group correlations (e.g., grouping *hairless* and *wings* to *Bat*), which can ease the biased prediction of unseen classes (e.g., grouping *furry* and *wings* to *Bird*). However, when learning complex knowledge, even humans may need to revisit information multiple times to progressively refine and rectify the learned knowledge for better understanding [15], [16]. These methods directly use attribute groups to refine localities without any calibration or rectification, which may not be able to learn complex semantics precisely.

On the other hand, a well-known education paradigm in cognitive theory, *Spiral Curriculum* [17], can teach adults and children to understand complex knowledge by revisits. This curriculum decomposes knowledge into a series of topics. Students first preview overall knowledge and build their views. Then, they gradually select some topics to review and revise until they fully understand. Motivated by this, we propose an alternative form of *Spiral Learning* for semantic learning. We take bird classification as an example. As shown in Fig. 1, when learning to distinguish an unseen class *Bird*, the conventional learning process (left) based on fixed attribute group learning may make biased predictions toward seen classes, e.g., *Bat* with the similar shapes and colors. Our proposed method (right) can preview images to conclude several possible attribute groups that may need further revisits. Then, we can progressively review our predictions by selecting suitable high-level semantics to learn (e.g., the yellow group in Fig. 1) and, thus, revise their predictions as *Bird*. This learning process allows models to group and select suitable attribute groups to spirally accumulate knowledge, which may ease the difficulty of learning semantic correlations of difficult classification tasks and reduce confusing information from

Manuscript received December 2, 2021; revised March 12, 2022; accepted May 15, 2022. *(Zhe Liu and Yun Li contributed equally to this work.) (Corresponding author: Zhe Liu.)*

Zhe Liu is with the Jiangsu Provincial Engineering Laboratory of Pattern Recognition and Computational Intelligence, Jiangnan University, Wuxi 214122, China, and also with the School of Computer Science and Engineering, University of New South Wales, Sydney, NSW 2052, Australia (e-mail: liuzhe960912@163.com).

Yun Li, Lina Yao, and Sam Dixon are with the School of Computer Science and Engineering, University of New South Wales, Sydney, NSW 2052, Australia (e-mail: yun.li5@student.unsw.edu.au; lina.yao@unsw.edu.au; z5351595@ad.unsw.edu.au).

Julian McAuley is with the School of Computer Science, University of California San Diego, La Jolla, CA 92093 USA (e-mail: jmcauley@eng.ucsd.edu).

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TNNLS.2022.3176282.

Digital Object Identifier 10.1109/TNNLS.2022.3176282

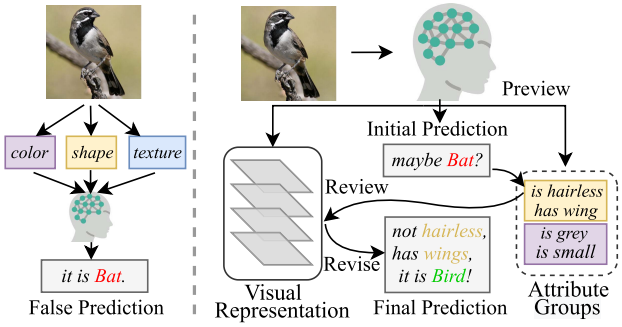

Fig. 1. Left: conventional methods directly learn images based on a series of fixed attribute groups, which may introduce some confusing information (e.g., color) without distinction. Right: proposed spiral process previews to group attributes and then progressively review predictions by learning suitable attribute groups with distinction.

insignificant semantics (e.g., the purple group) to ease the difficulty of learning complex semantic correlations.

In this article, we propose a two-stage reinforced self-revised (RSR) framework to implement spiral learning in an end-to-end manner. In the preview stage, we design a weakly supervised self-directed grouping function to automatically group attributes into high-level semantic groups. In the review stage, we propose a reinforced selection module and a revision module to simulate the process that dynamically selects attribute groups to revisit visual information. Different from conventional methods that learn visual localities and directly aggregate them with global knowledge, spiral learning aims to learn semantic localities from a global visual representation and, thus, progressively calibrate the learned knowledge. We summarize our contributions as follows.

1) We propose a novel RSR framework to decompose conventional learning processes as an incremental spiral learning process. By spirally learning a series of semantic localities based on attribute groups, RSR can dynamically revise the learned correlations and thus ease the difficulty of learning complex correlations.
2) We demonstrate our consistent improvement over state-of-the-art algorithms on four benchmark datasets in both zero-shot and generalized zero-shot settings. To show the extensibility of our framework, we also present an adversarial extension to boost simulation ability.
3) We conduct quantitative analysis as well as visualization of RSR, which indicates that our model can effectively find significant attributes and combine high-level semantic groups as insightful attribute groups to revise predictions in an incremental way. Moreover, the decision processes are explainable.

II. Related Work

A. Zero-Shot Learning

The key insight of ZSL is to capture common semantic correlations among both seen and unseen classes. For example, Yun *et al.* [18] propose a salient attribute learning network to explore and generate the characterized semantic embedding based on the class-specific information from the dimension space, which can learn more expressive visual information. Yang *et al.* [19] research the zero-shot localization problem by a semantic-assisted location network, which uses an expectation-maximization algorithm to ease the information loss caused by the inconsistency between image embedding space and class embedding space. A typical approach is to project visual and/or attribute features to a unified domain and then apply a compatibility function for classification [9], [20]–[23]. Non-end-to-end approaches [6]–[8] disentangle attributes or generate instances in the embedding space to ease the semantic and visual mismatch. More closely related to our work, modern end-to-end models [4], [5], [24], [25] are proposed to extract diverse visual localities corresponding to semantics and, thus, obtain the overall semantic correlation. Zhu *et al.* [24] propose a multiattention model to obtain multiple discriminative localities under semantic guidance. Xie *et al.* [5] further incorporate second-order embeddings to enable stable locality collaboration. Some works [4], [11]–[13], [26] propose to use attribute groups to enhance semantic learning. Atzmon and Chechik [11] and Jayaraman *et al.* [12] group attributes to find joint probabilities of individual attributes for precise prediction. Liu *et al.* [26] use class-specific attribute groups as weights to modify the layerwise outputs. Xu *et al.* [4] and Long and Shao [14] manually group attributes to regularize semantic learning. However, these works directly group attributes for final predictions or rely on extra manual group annotations, which cannot provide multiple complementary attribute groups and may fail to learn groups outside human definitions, e.g., a combined group of multiple manual attribute groups. In our work, we automatically group attributes into diverse groups, which can transcend human-defined groups. We learn a series of semantic localities that complement each other during spiral learning.

B. Reinforcement Learning in Related Fields

Reinforcement learning has been extensively investigated for object detection [27], [28], image classification [29], [30], and few-shot learning [31]. Mathe *et al.* [27] and Pirinen *et al.* [28] use reinforcement learning to improve sampling visual regions in an efficient and accurate way. Wang *et al.* [29] and Chu *et al.* [31] propose to focus on different visual regions of images and then aggregate regional information to obtain an enhanced overall judgment. Chen *et al.* [30] use a recurrent reinforced module to narrow down the visual space based on the spatial contextual dependence of visual regions. Moreover, Dogru *et al.* [32] develop a real-time object-tracking algorithm for the industry of oil sands, which can combine reinforcement learning and computer vision to synchronize control theory. Singh and Zheng [33] propose a semantic guidance pipeline to discover and combine the distinct position and scale information with the semantic localization and spatial information of instances. The pipeline proposes a semantic focus reward for agent training, enabling the reinforced agents to require any human supervision. These methods aim to use reinforced modules to sample visual regions from visual inputs and,

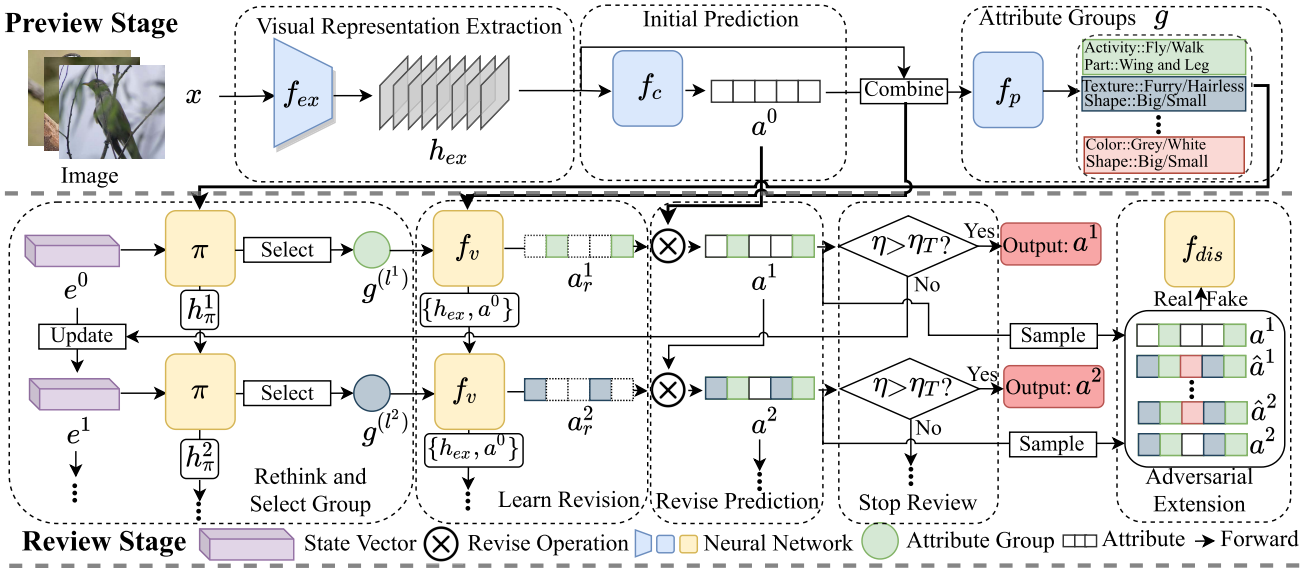

Fig. 2. Model overview. Given an instance x, RSR spirally learns and revises predictions of the class attribute for x. The *preview* stage (upper) extracts a visual representation h_{ex} by extractor f_{ex} and makes initial prediction a^0 by preview classifier f_c. A self-directed grouping function f_p learns a series of attribute groups as g. The *review* stage (lower) spirally revises previous predictions following a sequence of 3R processes. A reinforced module π dynamically rethinks and selects $\{(l^1)\text{th}, (l^2)\text{th}, \ldots\}$ attribute group in g. The revision module f_v revisits visual representation based on the selected attribute group to learn revisions $\{a_r^1, a_r^2, \ldots\}$. The model uses revisions to revises a^0 as $\{a^1, a^2, \ldots\}$ until the model obtains a confident prediction (i.e., $\eta > \eta_T$) or selects all groups. An optional adversarial extension samples and fuses ground-truth attributes $\{\hat{a}^1, \hat{a}^2, \ldots\}$ with $\{a^1, a^2, \ldots\}$ for an attribute discriminator f_{dis} to enable the model to simulate the prior attribute distribution.

thus, narrow down visual space to learn semantic information within regions. Our method is designed to rethink and select the most appropriate attribute groups to guide the learning process, which learns semantic information from the visual inputs in a global view governed by the selected attribute groups.

III. RSR Network

Problem Formulation: Given an instance $x \in \mathbb{R}^{H\times W\times C}$ with size $H \times W$ in C channels, each instance has class $y \in \mathbb{N}$ and the corresponding class attribute $\phi(y) = a \in \mathbb{R}^{1\times m}$ with m criteria. Let X, Y, and A be the sets of instances, ground-truth class labels, and predefined class attribute vectors, respectively. We define $\mathcal{S} = \{(x, y, a)|x \in X^S, y \in Y^S, a \in A^S\}$ and $\mathcal{U} = \{(x, y, a)|x \in X^U, y \in Y^U, a \in A^U\}$ as a training set (i.e., seen classes) and a testing set (i.e., unseen classes), respectively. Note that seen classes and unseen classes are disjoint, i.e., $\mathcal{S} \cap \mathcal{U} = \emptyset$, but A^S and A^U share the same criteria to allow knowledge transfer. Given a test instance (x, y, a), ZSL only predicts unseen instances, i.e., $(x, y, a) \in \mathcal{U}$; generalized ZSL (GZSL) predicts both seen and unseen instances, i.e., $(x, y, a) \in \mathcal{S} \cup \mathcal{U}$.

RSR network consists of two stages: **Preview** and **Review**. The preview stage previews instances and learns attribute groups. The review stage consists of *Rethink*, *Revisit*, and *Revise* (3R) processes. The review stage spirally learns semantic localities based on attribute groups to revise decisions in an incremental way. A model overview is shown in Fig. 2.

A. Preview Stage

The preview stage contains three modules: an information extractor f_{ex} for visual representation h_{ex}, a preview classifier f_c for an initial prediction a^0, and a self-directed grouping function f_p for attribute groups g.

1) Visual Representation Extraction and Preview Prediction: Spiral learning incrementally revisits information based on different attribute groups. To reduce the computation cost during revisits, we extract a visual representation h_{ex} for reuse. Considering that different attribute groups may correspond to different parts of inputs, we learn incompletely compressed embeddings by a convolutional neural network (CNN): $f_{\text{ex}}(x) \rightarrow h_{\text{ex}}$, which can keep the location information. Then, to obtain an initial preview prediction, we use a fully connected network (FCN) as a preview classifier: $f_c(h_{\text{ex}}) \rightarrow a^0$. We can jointly optimize f_c and f_{ex} by a preview cross-entropy loss $\mathcal{L}_{\text{PRE}}$

$$\min_{f_{\text{ex}}, f_c} \mathcal{L}_{\text{PRE}} = -\log \frac{\exp\big(f_c(f_{\text{ex}}(x))^T \phi(y)\big)}{\sum_{\hat{y}\in Y^S} \exp\big(f_c(f_{\text{ex}}(x))^T \phi(\hat{y})\big)} \tag{1}$$

where $h_{\text{ex}} \in \mathbb{R}^{14\times 14\times 1024}$ denotes the visual representation; $f_c(f_{\text{ex}}(x)) \rightarrow a^0$ denotes the initial prediction of the preview; and y and $\phi(y)$ denote the ground-truth label and the true attribute vector of x, respectively. $\mathcal{L}_{\text{PRE}}$ supervises h_{ex} and a^0 to learn the correct semantic correlations from a global perspective. a^0 can be viewed as an indicator of the learned global semantic correlations. The corresponding details about shape transformation of inputs can be found in Appendix A.

2) Grouping Attributes: Inspired by *Spiral Curriculum* [17] that splits complex concepts into several subconcepts, we decompose the overall attribute criteria into k diverse subgroups, i.e., attribute groups, and, thus, construct different tendencies for semantic learning. We adopt a CNN to project h_{ex} into 2048-D vectors and then combine h_{ex} with a^0 as

inputs for f_p, indicating the visual information and the preview learning state, respectively. We use an FCN as the self-directed grouping function $f_p(h_{\text{ex}}, a^0) \rightarrow \mathbb{R}_+^{k\times m}$ to find k diverse attribute groups as the potential semantic biases that need to be reviewed in a^0. f_p predicts k different weights for each criterion and reshapes the weights followed by a rectified linear unit (ReLU) to deactivate insignificant criteria as k subparts of all criteria, i.e., $g \in \mathbb{R}^{km} \rightarrow \mathbb{R}_+^{k\times m}$. We summarize f_p as follows [11]:

$$g = f_p\left(h_{\text{ex}}, a^0\right) = \text{ReLU}\left(\text{Reshape}\left(\omega_p\left(h_{\text{ex}}, a^0\right) + b_p\right)\right) \tag{2}$$

where ω_p and b_p denote the weight and bias parameters of f_p, respectively. We optimize f_p based on the spiral reviews in the **review** stage, which enables f_p to discover complementary attribute groups.

B. Review Stage

The review stage is a sequential process composed of 3R to progressively review and revise a^0 as $\{a^1, a^2, \ldots, a^t : t \leq k\}$ without repeated attribute groups, which is conducted by a reinforced selection module π and a revision module f_v. We first introduce review details and then the optimization of π.

The **rethink** process progressively selects suitable attribute groups to review visual information and eases biased semantic learning in previous predictions. Considering that the selection is progressive, and the attribute groups are fixed in the review stage, we design $e^t = \{h_{\text{ex}}, a^0, a^t\}$ as the tth state vector for π, where $\{h_{\text{ex}}, a^0\}$ can be viewed as the indicator of g and overall biased semantics; a^t represents the current state and indicates the solved biases. To enable π to progressively "rethink" based on the previous decisions, we use a gated recurrent unit (GRU) in π to maintain the previous hidden states h_π. We initialize the state vector as $e^0 = \{h_{\text{ex}}, a^0, a^0\}$ and the hidden state as $h_\pi^0 = \emptyset$. "Rethink" can be summarized: $\pi(e^t, h^t) \rightarrow l^{t+1} \in [1, k]$, which is a number to indicate the (l^{t+1})th group in g.

The **revisit** process learns the revision that extracts semantic information based on attribute groups. Given the selected (l^{t+1})th group in g, we first use elementwise multiplication to mask a^0 with the attribute group to highlight the target semantic locality. Then, we use an FCN as the revision module f_v to extract and refine the revision from the visual information h_{ex} by

$$\begin{aligned} a_r^{t+1} &= f_v\left(a^0, h_{\text{ex}}, g^{(l^{t+1})}\right) \\ &= \left(\omega_v\left(\left\{g^{(l^{t+1})} \odot a^0, h_{\text{ex}}\right\}\right) + b_v\right) \odot g^{(l^{t+1})} \end{aligned} \tag{3}$$

where a_r^{t+1} denotes the refined revision by multiplying the selected attribute group $g^{(l^{t+1})}$; $\odot$ denotes elementwise multiplication; and ω_v and b_v are learnable parameters of f_v. To enable a_r^{t+1} to learn semantic locality that can enhance correlation learning, we use a local cross-entropy loss $\mathcal{L}_{\text{LOC}}$

$$\mathcal{L}_{\text{LOC}} = -\log \frac{\exp\left(\left(a_r^{t+1}\right)^T \phi(y)\right)}{\sum_{\hat{y}\in Y^S} \exp\left(\left(a_r^{t+1}\right)^T \phi(\hat{y})\right)} \tag{4}$$

where y is the ground-truth label and $\phi(y)$ denotes the true attribute vector of the input instance. The masked revision deactivates insignificant attribute criteria, so only a part of the remaining criteria with large weights in a_r^{t+1} will significantly influence $\mathcal{L}_{\text{LOC}}$. In other words, $\mathcal{L}_{\text{LOC}}$ only optimizes a_r^{t+1} to learn generalized semantic locality of the highlighted attribute group.

The **revise** process fuses the revision a_r^{t+1} with the current prediction to enhance semantic learning. Given the current prediction vector a^t and the revision vector a_r^{t+1}, we propose to revise

$$a^{t+1} = \frac{a^t}{\|a^t\|} + \beta \frac{a_r^{t+1}}{\left\|a_r^{t+1}\right\|} \tag{5}$$

where $\beta = 1/(t+1)$ is an autoweighted factor to adjust the influence of the revision on the prediction.

We predict labels by finding the most similar class attribute by cosine similarity. We adopt cosine similarity to measure the similarities between the learned attributes and the normalized prior attributes (i.e., unit attribute vectors). Given a^t and a_r^{t+1}, we let $f_{\cos}(a^t, \phi(y))$ and $f_{\cos}(a_r^{t+1}, \phi(y))$ be the similarities of a^t and a_r^{t+1} to class y, respectively. Then, the similarity of the revised a^{t+1} to class y can be $f_{\cos}(a^{t+1}, \phi(y)) = (1/(\|a^{t+1}\|)) f_{\cos}(a^t, \phi(y)) + (\beta/(\|a^{t+1}\|)) f_{\cos}(a_r^{t+1}, \phi(y))$, where $f_{\cos}$ is the cosine similarity and $\phi(y)$ is the normalized true attribute. The corresponding proof is in Appendix D.

$f_{\cos}(a^{t+1}, \phi(y))$ is a weighted similarity of $f_{\cos}(a^t, \phi(y))$ and $f_{\cos}(a_r^{t+1}, \phi(y))$, which only propagates the similarity information in revision a_r^{t+1} to revise a^t. In other words, a^{t+1} is the revised prediction of a^t by the refined locality a_r^{t+1}. To supervise revisions and predictions to complement each other, we use an overall cross-entropy loss $\mathcal{L}_{\text{OA}}$ and a joint loss function $\mathcal{L}_{\text{JNT}}$ (6) and (7), as shown at the bottom of the page, where $(a^0)_i$, $(a^{t+1})_i$ denote prediction probabilities of the ith criterion in a^0 and a^{t+1} by cosine similarity, respectively; m is the attribute criterion dim. $\mathcal{L}_{\text{JNT}}$ optimizes the joint probability of a^0 and a^{t+1}. We take $\mathcal{L}_{\text{JNT}}$ as a regularization term to regularize locality learning to be consistent with the

$$\mathcal{L}_{\text{OA}} = -\log \frac{\exp\left(\left(a^{t+1}\right)^T \phi(y)\right)}{\sum_{\hat{y}\in Y^S} \exp\left(\left(a^{t+1}\right)^T \phi(\hat{y})\right)} \tag{6}$$

$$\mathcal{L}_{\text{JNT}} = -\log \left\{ \frac{\sum_{i\in[1,m]} \left[\exp\left(\left(a^0\right)_i^T \phi(y)\right)\right]\left[\exp\left(\left(a^{t+1}\right)_i^T \phi(y)\right)\right]}{\left[\sum_{\hat{y}\in Y^S} \exp\left(\left(a^0\right)^T \phi(\hat{y})\right)\right]\left[\sum_{\hat{y}\in Y^S} \exp\left(\left(a^{t+1}\right)^T \phi(\hat{y})\right)\right]} \right\} \tag{7}$$

global prediction by modifying the influence of attributes to the optimization based on a^0.

Then, we can learn the sequence of revised predictions $\{a^1, a^2, \ldots, a^t\}$ and the corresponding revisions $\{a_r^1, a_r^2, \ldots, a_r^t\}$. We can summarize the unified review loss function $\mathcal{L}_{\text{REV}}$ as follows:

$$\min_{f_p, f_v} \mathcal{L}_{\text{REV}} = \sum_t \alpha^{t-1} \left[\mathcal{L}_{\text{LOC}}(a_r^t) + \mathcal{L}_{\text{OA}}(a^t) + \mathcal{L}_{\text{JNT}}(a^0, a^t) \right] \tag{8}$$

where $\alpha \in (0, 1)$ is a predefined discount parameter. $\mathcal{L}_{\text{REV}}$ enables the review processes to learn refined semantic localities to complement each other in an incremental way, which optimizes attribute groups in g to be constantly linked to diverse and different semantics.

The **reinforced selection module** is designed to enhance the review process with better group selection. Thus, π aims to improve the accuracy of predicting the ground-truth label y during the review stage. To measure the improvement in accuracy, we define $p(\mathcal{L})_i$ as the prediction probability of class i in loss $\mathcal{L}$. The goal of π is to improve $\overline{p(\mathcal{L}_{\text{REV}})_y}$, i.e., the mean correct probability of the terms in $\mathcal{L}_{\text{REV}}$. To highlight the revision significance during selection, we define $R_{\text{RSR}} = \overline{[p(\mathcal{L}_{\text{REV}})_y + p(\text{UNI})_y]}$ as the reward of π, where $p(\text{UNI})_y = [p(a^0)_y + \sum_t (1/(1+t)) p(a_r^t)_y]$ is a union probability of the initial prediction and unnormalized revisions to enlarge the probability difference. Then, we optimize π by maximizing

$$\max_{\pi} \mathbb{E}\left[\sum_t \gamma^{t-1} R_{\text{RSR}} \right] \tag{9}$$

where $\gamma \in (0, 1)$ is a predefined discount parameter. We implement this optimization by proximal policy optimization (PPO) following [34]. See Appendix E for more optimization details.

The **confidence parameter** enables our model to autohalt the review stage. We define a confidence parameter from the perspective of achieving a solid union prediction $\eta = \max_i [p(\text{UNI})_i]$, which is the highest label probability in the union probability. Given a predefined probability threshold η_T, our model early stops the review stage when obtaining a confident prediction, i.e., $\eta > \eta_T$.

C. Adversarial Training Extension

Similar to adversarial training [35], a crucial property of ZSL is to simulate the same semantic correlation in the real semantic distribution. Therefore, adversarial training may enhance the model by using prior attribute distributions to regularize the learned attribute distribution. This section introduces an adversarial version of RSR (A-RSR), which implements adversarial training using the same structure but an additional attribute discriminator f_{dis}. Considering A-RSR as an extractor, we use $f_{\text{dis}}(a^t) \to [0, 1]$ to distinguish attributes, where "0" denotes the fake attribute from A-RSR and "1" is the true attribute. Then, we optimize an adversarial loss function $\mathcal{L}_{\text{A-RSR}}$ to regularize the model to simulate the prior attribute distribution by confusing f_{dis}

$$\begin{aligned} \min_{f_{\text{ex}}, f_c, f_p, f_v} \max_{f_{\text{dis}}} \sum_t \alpha^{t-1} \big\{ & \mathbb{E}_{\hat{a}^t \sim A^S} \left[\log f_{\text{dis}}(\hat{a}^t) \right] \\ & + \mathbb{E}_{a^t \sim \text{A-RSR}(x)} \left[\log(1 - f_{\text{dis}}(a^t)) \right] \\ & + \mathcal{L}_{\text{RSR}}(a^t) \big\} \end{aligned} \tag{10}$$

where α is the same discount parameter in RSR; $\hat{a}^t \sim A^S$ is the ground-truth attribute from the prior attribute distribution of inputs; a^t is an attribute from the learned attribute distribution; and $\mathcal{L}_{\text{RSR}} = \mathcal{L}_{\text{PRE}} + \mathcal{L}_{\text{REV}}$ is an auxiliary classifier loss to optimize the spiral learning modules. Then, we split the optimization of π into two components to better serve adversarial training. Following (9), we redesign reward functions as $R_{\text{DIS}} = [1 - f_{\text{dis}}(a^t)]$, $R_{\text{A-RSR}} = [R_{\text{RSR}} + f_{\text{dis}}(a^t)]$ for training and confusing f_{dis}, respectively. Both R_{DIS} and $R_{\text{A-RSR}}$ enable π to find the most significant semantic groups, which can assist or constrain f_{dis}. Thus, we can optimize (9) based on the new rewards to enable π to be consistent with learning goals of (10). Note that we do not use $\hat{a}^t$ for training π to let π focus on capturing the significant semantics for A-RSR.

D. Implementation Details

1) Training Strategy: We propose a two-stage training procedure to ease the training difficulty. **Stage-I** optimizes the nonreinforced modules to enable models to provide reliable judgment. We first optimize f_{ex} and f_c in the preview stage to extract the reliable visual representation and initial prediction using (1). We pretrain the model to provide a set of initial solid parameters for the following reinforcement training, preventing training collapse due to the complex training process. Then, we fix parameters of f_{ex} and f_c. We use random group selection to optimize f_p, f_v, and f_{dis} without early stopping to be capable of handling the general review situations following (8) and (10) for RSR and A-RSR, respectively. We use random group selection to replace the reinforced selection module at first to mimic the learning process of humans. Humans will make random attempts in the beginning stage of learning knowledge, which will help models accumulate more generalized knowledge during training. **Stage-II** optimizes the reinforced selection module. We fix the nonreinforced modules and use π to select locations with early stopping, which enables our model to be able to handle the general selection situation and precisely autohalt in the inference stage. PPO optimizes π to maximize reward functions using (9) based on the corresponding rewards for RSR and A-RSR, respectively.

2) Inference Strategy: The model autohalts the proceeding once $\eta > \eta_T$ or all groups have been selected, i.e., $t = k$. Given an arbitrary revised prediction a^t, we predict labels as follows:

$$\begin{aligned} \text{ZSL:} \quad & \hat{y} = \arg\max_{\hat{y} \in Y^U} (a^t)^T \phi(\hat{y}) \\ \text{GZSL:} \quad & \hat{y} = \arg\max_{\hat{y} \in Y^U \cup Y^S} \left[(a^t)^T \phi(\hat{y}) - \varepsilon \delta(\hat{y}) \right] \end{aligned} \tag{11}$$

where $\hat{y}$ denotes the predicted label; $\varepsilon \in [0, 1]$ is a calibration factor to fine-tune the model toward unseen classes. δ is a sign function that returns 1 if $\hat{y} \in Y^U$ or 0 otherwise. The second

term in GZSL is calibrated stacking [36], which is commonly used in end-to-end models [4], [5] to prevent models being largely biased toward seen classes due to the lack of training instances of unseen classes.

IV. Experiments

We validate our models on four widely used benchmark datasets: Scene UNderstanding (SUN) [37], Caltech-UCSD Birds (CUB) [38], Attributes Pascal and Yahoo (aPY) [39], and Animals With Attributes (AWA2) [40].

SUN [37] is a comprehensive dataset of annotated images covering a large variety of environmental scenes, places, and objects. The dataset consists of 14 340 fine-grained images from 717 different classes. Following [40], the dataset is divided into two parts to prevent overlapping unseen classes from the ImageNet classes: 645 seen classes for training and the rest 72 unseen classes for testing. The attribute is human-annotated and is of 102 dimensions.

AWA2 [40] is a subset of the AWA dataset, which is an updated version of AWA1. The AWA2 is the only dataset provided with the source images, while the raw images of the AWA1 dataset are not provided. This dataset contains 37 322 images from 50 animal species captured in diverse backgrounds. AWA2 selects 40 classes for training and ten classes for testing in the ZSL setting. The AWA dataset is annotated with binary and continuous attributes, and we take the 85-D continuous attributes for the experiments following [40], which is more informative.

CUB [38] consists of 11 788 images from 200 different bird species. CUB is a fine-grained dataset that some of the birds are visually similar, and even humans can hardly distinguish them. It is challenging that a limited number of instances are provided for each class, which only contains nearly 60 instances. CUB splits classes as 150/50 for train/test in the ZSL setting.

aPY [39] comprises of 15 339 images from 32 classes. In the ZSL setting, 20 classes are viewed as seen classes for training, and the remaining ten classes are used as unseen classes for testing. The dataset is annotated with 64-D attributes. This dataset is very challenging that the classes are very diverse. aPY constitutes two subsets—aYahoo images and Yahoo aPascal images—so there may exist similar objects with different class/attribute semantic correlations.

The datasets are divided by proposed split (PS) to prevent overlapping train/test classes [40]. For the comprehensive comparison, we compare our model with 15 state-of-the-art methods, including six locality-based methods (denoted by *). We report these methods in the inductive mode [40] without manual group side information [4] during training. We use stochastic gradient descent (SGD) [41] to train our models. We set α as 0.9 and γ as 0.99 on four datasets. Grid search (in Section IV-C) is used to set η_T as 0.4 and k as 5, except $\eta_T = 0.7$ on aPY for A-RSR. See Appendix A for more parameter and architecture details.

A. Main Results of Zero-Shot and GZSL

To validate the effectiveness of the proposed modules, we take the preview stage as the baseline (the base model), which is a fine-tuned ResNet101 [46] with a classifier. Then, we show the results on the best step during group selections of variants w/o reinforced module or adversarial extension: RSR, Random-self-revised (SR), Random-adversarial SR (ASR), and A-RSR, respectively. See Section IV-C for accuracy per step. Note that Random-ASR and A-RSR use Random-SR as the pretrained backbone. In Table I, we measure average per-class Top-1 accuracy (T1) in ZSL, Top-1 accuracy on seen/unseen (s/u) classes, and their harmonic mean (H) in GZSL.

In Table I, compared with the *base model*, our framework improves the initial prediction by 4.2%/5.9% (SUN), 4.0%/4.0% (CUB), 6.0%/6.9% (aPY), and 2.1%/2.8% (AWA2) in ZSL/GZSL. The progress of Random-SR and Random-ASR proves spiral learning to be effective at revising the preview predictions with semantic localities refined by attribute groups. Compared with random selection, RSR and A-RSR further improve random selection by up to 1.4%/1.9% in ZSL/GZSL, demonstrating that the self-directed grouping function can provide complementary attribute groups and reinforced selection better uses the group relationships. The adversarial extension slightly impairs the performance of Random-ASR on AWA2. This may be caused by the significant domain shift between seen and unseen classes, which may lead to the model simulating biased seen distributions. Otherwise, the adversarial extension improves performance by up to 1.2% in both ZSL and GZSL. The adversarial training enhances the performance of Random-SR more than the performance of RSR, which is because the reinforced selection may provide too specific visual information and, thus, lack generalization for adversarial training to learn.

Compared with state-of-the-art algorithms, we can observe that the most basic spiral learning model, i.e., Random-SR, can obtain the state-of-the-art performance of locality-based methods, which demonstrates the advantage of learning refined semantic localities based on attribute groups. The proposed reinforced selection module further improves the performance: RSR (e.g., CUB and AWA2) and A-RSR (e.g., SUN and aPY) achieve the best performance, which demonstrates the effectiveness of spiral learning in tackling complex correlations. Spiral learning can review different attribute groups and revisit visual information to spirally understand the correlations that cannot be understood with one-time learning. RSR and A-RSR consistently outperform other methods by a large margin, i.e., 2.7%/2.8%, 0.6%/0.8%, and 4.5%/1.8% on SUN, CUB, and aPY in ZSL/GZSL, respectively. Our models obtain the highest unseen scores and harmonic mean accuracy on four datasets, which demonstrates our effectiveness in learning unbiased localities to revise the learned global semantic correlation. The improvement of performance on CUB and AWA2 is not as significant as that on SUN and aPY, which may be caused by the low accuracy of finding significant attributes and the sparse attribute criterion weights of the learned groups, which will be discussed in Section IV-B.

B. Attribute Group and Decision Process Analysis

In this section, we conduct a quantitative analysis on the learned attribute groups to demonstrate the effectiveness of the self-directed grouping function, and we visualize the decision process to illustrate the strong explainability of the decision

TABLE I

MAIN RESULTS. * DENOTES LOCALITY-BASED METHODS. THE BASE MODEL USES a^0 AS PREDICTIONS. WE MEASURE AVERAGE PERCLASS ACCURACY OF TOP-1 (T1), UNSEEN/SEEN CLASSES (u/s), AND HARMONIC MEAN (H)

Method	ZSL				GZSL											
	SUN	CUB	aPY	AWA2	SUN			CUB			aPY			AWA2		
	T1	T1	T1	T1	u	s	H	u	s	H	u	s	H	u	s	H
Non End-to-End																
SP-AEN[6]	59.2	55.4	24.1	58.5	24.9	**38.6**	30.3	34.7	70.6	46.6	13.7	63.4	22.6	23.0	90.9	37.1
RelationNet[21]	-	55.6	-	64.2	-	-	-	38.1	61.1	47.0	-	-	-	30.0	**93.4**	45.3
PSR[42]	61.4	56.0	38.4	63.8	20.8	37.2	26.7	24.6	54.3	33.9	13.5	51.4	21.4	20.7	73.8	32.3
*PREN[20]	60.1	61.4	-	66.6	35.4	27.2	30.8	35.2	55.8	43.1	-	-	-	32.4	88.6	47.4
Generative Methods																
cycle-CLSWGAN[7]	60.0	58.4	-	67.3	47.9	32.4	38.7	43.8	60.6	50.8	-	-	-	56.0	62.8	59.2
f-CLSWGAN[8]	58.6	57.7	-	68.2	42.6	36.6	39.4	43.7	57.7	49.7	-	-	-	57.9	61.4	59.6
TVN[43]	59.3	54.9	40.9	68.8	22.2	38.3	28.1	26.5	62.3	37.2	16.1	**66.9**	25.9	27.0	67.9	38.6
Zero-VAE-GAN[44]	58.5	51.1	34.9	66.2	44.4	30.9	36.5	41.1	48.5	44.4	30.8	37.5	33.8	56.2	71.7	63.0
ResNet101-ALE[45]	57.4	54.5	-	62.0	20.5	32.3	25.1	25.6	64.6	36.7	-	-	-	15.3	78.8	25.7
End-to-End																
QFSL[9]	56.2	58.8	-	63.5	30.9	18.5	23.1	33.3	48.1	39.4	-	-	-	52.1	72.8	60.7
*SGMA[24]	-	71.0	-	68.8	-	-	-	36.7	71.3	48.5	-	-	-	37.6	87.1	52.5
*LFGAA[26]	61.5	67.6	-	68.1	20.8	34.9	26.1	43.4	**79.6**	56.2	-	-	-	50.0	90.3	64.4
*AREN[5]	60.6	71.5	39.2	67.9	40.3	32.3	35.9	63.2	69.0	66.0	30.0	47.9	36.9	54.7	79.1	64.7
*SELAR-GMP[25]	58.3	65.0	-	57.0	22.8	31.6	26.5	43.5	71.2	54.0	-	-	-	31.6	80.3	45.3
*APN[4]	60.9	71.5	-	66.3	41.9	34.0	37.6	65.3	69.3	67.2	-	-	-	56.5	78.0	65.5
Ours																
Base model (preview)	60.0	68.1	39.4	66.9	43.4	28.6	34.5	61.7	66.4	64.0	34.2	29.7	31.8	**58.4**	68.0	62.8
*Random-SR	63.1	71.7	43.2	68.9	**51.0**	30.8	38.4	62.8	72.9	67.5	29.9	48.1	36.9	55.4	74.8	63.7
*RSR	64.0	**72.1**	44.2	**69.0**	49.8	32.0	39.0	63.5	73.0	**68.0**	**31.8**	46.1	37.6	56.0	79.1	**65.6**
*Random-ASR	63.8	71.8	44.0	68.4	49.1	33.9	40.1	**65.5**	69.7	67.5	30.1	49.6	38.1	56.8	74.0	64.3
*A-RSR	**64.2**	72.0	**45.4**	68.4	48.0	34.9	**40.4**	62.3	73.9	67.6	31.3	50.9	**38.7**	55.3	76.0	64.0

TABLE II

ATTRIBUTE-LEVEL ANALYSIS ON g. A HIGH SPARSITY DEGREE INDICATES WEIGHT IMBALANCE

Method	Top-10 shot accuracy				Sparsity degree			
	SUN	CUB	aPY	AWA2	SUN	CUB	aPY	AWA2
RSR	0.821	0.496	1.00	0.783	1.865	2.230	2.782	3.164
A-RSR	0.816	0.487	1.00	0.685	1.921	2.166	2.272	4.467

process. We analyze the components of g based on the human annotation to discover some cognitive insights. The analysis is conducted from three perspectives: the attribute level, the group level, and the decision level. The criterion number differs in different groups, so we take the maximum 10% weighted attributes to represent the group tendencies of learning semantics.

1) Attribute-Level Analysis: We first analyze attribute groups from an attribute perspective, i.e., finding significant attribute criteria and learning a balanced weight distribution. To analyze criterion significance, we view the top-ten largest criteria as the most significant criteria annotated by humans because we use the feature variance as attribute vectors [40] (a larger value indicates more significant features). Then, given an instance set X, we calculate the average *top-ten shot accuracy* to measure whether g contains the significant attribute criteria by $[\sum_{x\in X}\psi(g_x)]/|X|$, where ψ denotes a sign function to return 1 if any top-ten largest attribute criterion $a_i \in g_x$, otherwise 0; $|X|$ denotes the instance number. In other words, we calculate the ratio of the number of instances that can correctly classify the important attributes in manual annotations as the most ten important attributes in g to the instance number in the dataset. To measure the balance of weight distribution, we calculate the ratio of the maximum weight to the minimum weight in attribute groups as the *sparsity degree*, i.e., $\max(g)/\min(g)$. When a large sparsity

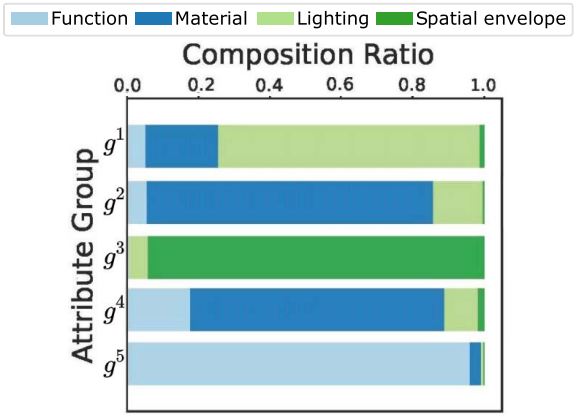

Fig. 3. Semantic analysis of learned attribute groups (g^1–g^5). We annotate the semantic groups (in legend) for attribute criteria in each attribute group based on the human annotations [37] to reveal the semantic meaning. We plot the relative composition ratios of semantics in each group to illustrate the diverse semantic tendencies of attribute groups.

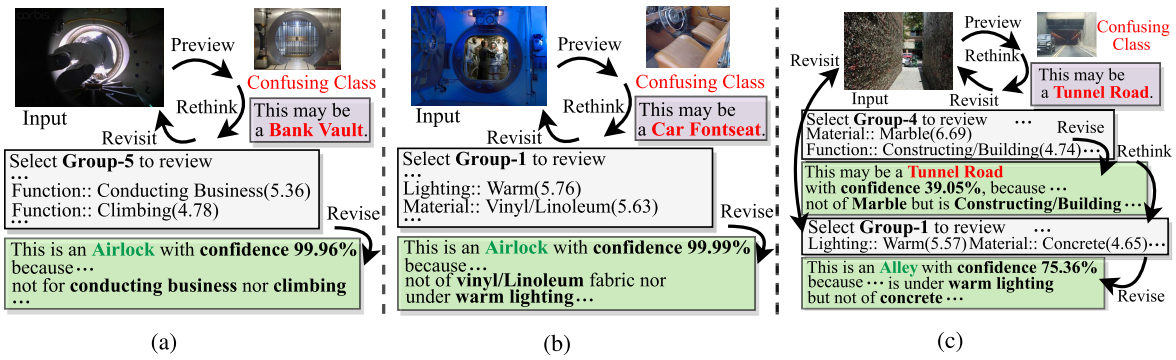

Fig. 4. Decision process visualization. Numbers in brackets are specific attribute criterion weights. (a) Instance-1. (b) Instance-2. (c) Instance-3.

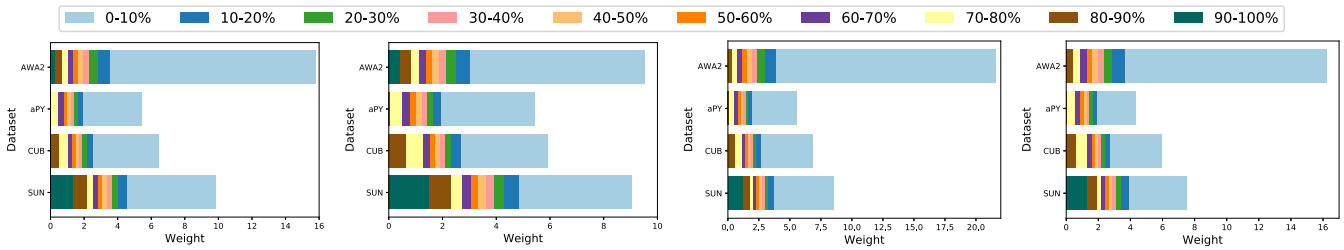

Fig. 5. Weight distributions during training and testing of RSR and A-RSR, respectively. The x-axis denotes the specific weight value, and the y-axis denotes different datasets. From left to right in the legend, we sort the weights from low to high and split them into groups with the same sizes of 10% nonzero attribute criteria. (a) RSR training stage. (b) RSR testing stage. (c) A-RSR training stage. (d) A-RSR testing stage.

degree exists in attribute groups, it indicates that criterion weights may be imbalanced. We summarize top-ten shot accuracy and sparsity degree in Table II. We can observe that the self-directed grouping function can effectively find significant criteria from the cognitive judgment of humans, especially aPY. The function can find few significant criteria on CUB, which may lead to the limited improvement of our framework. Criterion weights are balanced on SUN, CUB, and aPY but are slightly imbalanced on AWA2. The high sparsity degree of A-RSR on AWA2 misguides the model to learn a local optimum.

2) Group-Level Analysis: To analyze semantic meaning of the learned attribute groups, we refer to human annotations from SUN [37], which splits attribute criteria into four high-level semantic groups: *function*, *material*, *lighting*, and *spatial envelope* (e.g., *warm* belonging to *lighting*). Note that we do not use manual group annotations during training. We calculate the relative composition ratios of attribute criteria, belonging to different manual semantic groups, to our learned attribute groups as semantic tendencies. See Appendix F for calculation details of composition ratios. In Fig. 3, we take five attribute groups (i.e., g^1-g^5) learned by RSR as examples and exhibit their average semantic tendencies on SUN. We can observe that the weakly supervised f_p can build five diverse semantic tendencies. For example, g^5 mainly focuses on *function*, while g^4 combines *material*, *function*, and *lighting* as an attribute group, which shows that g can provide insightful attribute groups transcending manual annotations, i.e., combined semantic groups.

3) Decision-Level Analysis: In Fig. 4, we visualize three representative instances of *airlock* and *alley* with the corresponding attribute names, e.g., climbing, from the SUN dataset [37]. From the first two *airlock* instances, we can observe that the reinforced module selects different groups for locality learning due to the different confusing classes. Our model revises the preview prediction *bank vault* and *car fontseat* by learning the business/climbing function, seat material, and lighting, respectively. Comparing instance-2 with instance-3, the self-directed grouping function produces different weights in warm lighting, which indicates that criterion weights are instance-specific. For instance-3 that contains complex correlations, the reinforced module selects suitable groups to progressively distinguish *tunnel road* and *alley* until finding *not of concrete* to make a confident prediction. This dynamic decision process exhibits the strong explainability of our framework.

C. Ablation Study

1) Attribute Criterion Weight Distribution Analysis: To explore the detailed weight distribution in attribute groups, we sort attribute criterion weights from low to high and split them into groups with the same sizes of 10% nonzero criteria in Fig. 5. We can observe that most of the weights are small, and the model will slightly modify them with the new revisited information. Only the top 10% criteria take the highest weights in the groups, showing the significant influences during the review stage and can be viewed as the representative semantic tendencies of attribute groups. By comparing the training and testing stages, we can find that the weight distributions of the

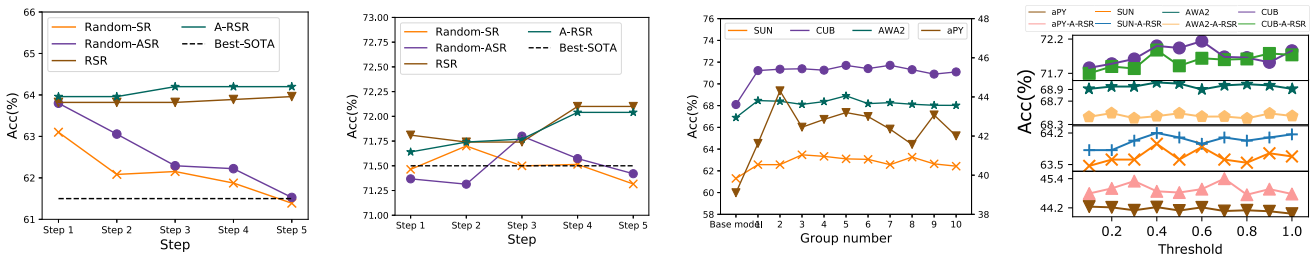

Fig. 6. (a) and (b) Accuracy curves. (c) and (d) Hyperparameter study on k and η_T, respectively.

TABLE III
ZSL RESULTS FOR NON-PRETRAINED A-RSR

Method	SUN	CUB	aPY	AWA2
Random-ASR	62.1	69.6	40.5	68.1
A-RSR	62.2	71.1	41.1	68.2

font 90% criteria in testing are sparser than those in training, indicating that it is more difficult for the self-directed grouping function to find and concentrate on the significant criteria of the unseen classes. By comparing RSR and A-RSR, we can observe that both models have relatively balanced weight distributions on SUN, which leads to good performance in ZSL and GZSL experiments. Also, RSR and A-RSR have the most imbalanced weight distributions on AWA2. While RSR shows better weight distribution in the testing stage than the training distribution, the testing weight distribution of A-RSR tends to be even sparser than in the training stage, which leads to the poor performance of A-RSR on AWA2. The imbalanced weight distribution may be caused by the significant domain shift in AWA2, which misguides the adversarial training to simulate the inaccurate semantic correlation.

2) Step-Accuracy Curve: Fig. 6(a) and (b) takes SUN and CUB as examples to show accuracy of each step. Compared with a random selection that cannot select suitable attribute groups, RSR and A-RSR can progressively improve predictions, especially on CUB. This indicates that the reinforced module can utilize complementary attribute groups to guide revisits to visual information and help models incrementally understand some complex correlations that are difficult to learn within a single step.

3) Hyperparameter Study: Fig. 6(c) exhibits k analysis of Random-SR, which is the initial parameters for variants. k largely influences Random-SR on aPY but less so on other datasets, which indicates that aPY can be easy to overfit. The best (on average) performance is obtained when $k = 5$. Fig. 6(d) exhibits η_T study of RSR and A-RSR. η_T is sensitive to A-RSR on aPY (i.e., ± 0.43) but stable on other datasets (i.e., ± 0.28). Overall, our models are stable under different parameters.

4) ZSL Performance of A-RSR Without Pretraining: In Table III, we show the A-RSR results without using a pretrained RSR backbone. We can find that the ZSL performance is significantly lower than the results using the pretrained RSR backbone, decreasing up to 2.0%, 2.2%, 4.3%, and 0.2% on SUN, CUB, aPY, and AWA2. The results indicate that the attribute discriminator is not able to directly regularize the spiral learning. When the base model does not have enough capability of learning an accurate prediction, the adversarial learning may not be able to optimize the base model to obtain overall optimality. Therefore, adversarial training should be a good optional extension of our model to achieve an enhanced spiral learning when the model is well-trained and the domain shift is not significant.

V. CONCLUSION

In this work, we present a spiral learning scheme inspired by Spiral Curriculum and propose an end-to-end RSR framework for ZSL. RSR spirally reviews visual information based on complementary attribute groups to learn the complex correlations that are difficult to capture without revisits. The consistent improvement on four benchmark datasets demonstrates the advantage of revisiting semantic localities. We validate the learned attribute groups from quantitative and explainable perspectives, which verifies that the weakly supervised self-directed grouping function is able to find significant attributes and insightful semantic groups. We also visualize the decision process to illustrate the explainability of the spiral learning. The adversarial extensibility of RSR shows promise for application in other ZSL settings, such as generative learning. In the future, we plan to apply our novel learning manner, i.e., the RSR framework based on spiral learning, in more diverse real-world scenarios, e.g., object detection and neural language processing, to enhance the ability to learn difficult tasks.

APPENDIX

A. More Implementation Details

We conduct experiments on Python 3.7.9 in Linux 3.10.0 with a GP102 TITANX driven by CUDA 10.0.130 with a fixed seed 272. The neural networks are implemented on Pytorch 1.7.0 and complied with GCC 7.3.0.

B. Parameter Setting

For calibrated stacking [36], we set ε to 1.2×10^{-6} on SUN, 1.12×10^{-5} on CUB, 2.9×10^{3} on aPY, and 1.2×10^{-1} on AWA2 tuned on a held-out validation set following [4], respectively. We use SGD [41] to train our model in an end-to-end manner with a momentum of 0.9, a weight decay of 10^{-5}, and a learning rate of 10^{-3}.

C. Network Architecture

Our framework consists of five modules: the visual representation extractor f_{ex}, the preview classifier f_c, the self-directed grouping function f_p, the revision module f_v, and the reinforced selection module π. We provide the source code of the testing stage with detailed network architecture codes in Supplementary files.

We let FCN(m) be an FCN layer with output dimension m, Max(7×7) be a max-pooling layer with the kernel size of 7×7, Adp(1×1) be an adaptive average pooling layer with the output kernel size of 1×1, Dropout(0.5) be a dropout layer with the keeping rate of 0.5, ReLU be the rectified linear activation function, and GRU(1024) be a GRU with the hidden state size of 1024. The parameters that we do not provide are set to the default values.

Visual representation extractor $f_{\text{ex}}(x) \rightarrow h_{\text{ex}}$ is a subnet of ResNet101 pretrained on ImageNet [47], which is composed of the blocks before the second-last layer before classifier in the original ResNet101.

Preview classifier $f_c(h_{\text{ex}}) \rightarrow a^0$ is composed of the second-last layer before classifier in the original ResNet101 pretrained on ImageNet, a pooling layer, an FCN layer, and a dropout layer. We use Max(7×7) on CUB and Adp(1×1) on other datasets as the pooling layer. Then, the output of the pooling layer is reshaped as 2048-D vectors. We adopt a two-layer FCN (i.e., FCN(1024) $-$ FCN(m)) on aPY and one-layer FCN (i.e., FCN(m)) on other datasets, where m denotes the attribute dim. Specifically, the one-layer FCN is without bias term on CUB. The dropout layer keeping rate is set to 0.7 on AWA2 and 0.5 on other datasets.

Self-directed grouping function $f_p(h_{\text{ex}}, a^0) \rightarrow \mathbb{R}_+^{k \times m}$ embeds h_{ex} using the second-last module of ResNet101 followed by Adp(1×1) and reshapes the output to 2048-D vectors. We concatenate the 2048-D vectors with a^0 and adopt a two-layer FCN, i.e., FCN($1024 + m/2$) $-$ ReLU $-$ FCN($k * m$) $-$ ReLU to obtain the attribute groups, where k is the predefined group number.

Revision module $f_v(a^0, h_{\text{ex}}, g^{l^t}) \rightarrow a_r^t$ adopts the same structure as f_p to embed h_{ex} into 2048-D vectors, i.e., the second-last module of ResNet101 followed by Adp(1×1). Then, we concatenate the vectors with the masked $g^{(l^t)} \odot a^0$ and feed them to an FCN layer to obtain the revision. We design the FCN layer as FCN($1024 + m/2$) $-$ FCN(m) $-$ Dropout on aPY and FCN(m) $-$ Dropout on other datasets. Similar to the design of f_c, we use FCN without bias term on CUB. The keeping rate of dropout layer is set to 0.7 on AWA2 and 0.5 on other datasets.

Reinforced module $\pi(e^t, h_\pi^t) \rightarrow l^t$ is a recurrent actor–critic network, which utilizes a recurrent module to extract information from e^t, h_π^t to feed actor and critic, respectively. First, we use the concatenated $\{h_{\text{ex}}, a^0\}$ as the component of e^t, which is composed of the compressed 2048-D h_{ex} and the preview prediction a^0 from f_p. We extract information of this part via an FCN layer, i.e., FCN(1024). Then, we use FCN(m, 256) $-$ ReLU $-$ FCN(512) if $m < 256$, otherwise FCN(512), to extract information from a^t. Finally, we concatenate the extracted information from $\{h_{\text{ex}}, a^0\}$ and a^t with h_π^t to feed GRU(1536) to let the output contain the previous selection information. With the output of GRU, we use FCN(512)$-$ReLU$-$FCN(k)$-$Softmax as actor module to calculate the probability distribution of all the locations and adopt FCN(512) $-$ ReLU $-$ FCN(1) as the critic module to infer the current state score, where k is the predefined group number, and the hidden state dim of GRU equals the input dimension.

D. Proof of Remark 1

Proof: Given the current prediction vector a^t and the revision vector a_r^{t+1}, we let $\phi(y)$ be the unit attribute vector of the ground-truth label y. Following [48], we can calculate the cosine similarities to $\phi(y)$ for a^t and a_r^{t+1} as follows:

$$f_{\cos}\left(a^t, \phi(y)\right) = \frac{\sum_{i \in [1,m]} a_i^t \phi(y)_i}{\|a^t\| \|\phi(y)\|} \tag{12}$$

$$f_{\cos}\left(a_r^{t+1}, \phi(y)\right) = \frac{\sum_{i \in [1,m]} \left(a_r^{t+1}\right)_i \phi(y)_i}{\left\|a_r^{t+1}\right\| \|\phi(y)\|}. \tag{13}$$

Similarly, we can easily have

$$\begin{aligned} & f_{\cos}\left(a^{t+1}, \phi(y)\right) \\ &= \frac{\sum_{i \in [1,m]} \left(\frac{1}{\|a^t\|} a^t + \frac{\beta}{\left\|a_r^{t+1}\right\|} a_r^{t+1}\right)_i \phi(y)_i}{\left\|a^{t+1}\right\| \|\phi(y)\|} \\ &= \frac{1}{\left\|a^{t+1}\right\|} f_{\cos}\left(a^t, \phi(y)\right) + \frac{\beta}{\left\|a^{t+1}\right\|} f_{\cos}\left(a_r^{t+1}, \phi(y)\right). \end{aligned} \tag{14}$$

E. Proximal Policy Optimization

Our reinforced selection module is implemented by a recurrent actor–critic network composed of an actor π and a critic V, where V aims to estimate the state value [34]. During the training process, we sample the location of the selected group l following $l^{t+1} \sim \pi(\text{loc}|e^t, h_\pi^t)$ to optimize the actor–critic network, where e^t denotes the state and h_π^t is the hidden state in the recurrent module for the tth step. We maximize a unified form of reward function for RSR and A-RSR as follows:

$$\max_\pi \mathbb{E}\left[\sum_t \gamma^{t-1} R^t\right] \tag{15}$$

where $\gamma = 0.99$ is a predefined discount parameter and R_t denotes the reward function for RSR or A-RSR. According to the work of Schulman *et al.* [34], the optimization problem can be addressed by a surrogate objective function using stochastic gradient ascent

$$\mathcal{L}_{\text{CPI}}^t = \frac{\pi\left(l^{t+1} \middle| e^t, h_\pi^t\right)}{\pi_{\text{old}}\left(l^{t+1} \middle| e^t, h_\pi^t\right)} \hat{f}_{\text{ad}}^t \tag{16}$$

where π_{old} and π represent the before and after updated policy network, respectively. $\hat{f}_{\text{ad}}^{t}$ is the advantages estimated by V as follows:

$$\hat{f}_{\text{ad}}^{t} = -V\left(e^{t}, h_{e}^{t}\right) + \sum_{t \le i \le k} \gamma^{i-t} R^{t} \tag{17}$$

where k denotes the maximum length of the sampled groups, i.e., the predefined group number. The optimization of π usually gets trapped in local optimality via some extremely great update steps when directly optimizing the loss function, so we optimize a clipped surrogate objective $\mathcal{L}_{\text{CPI}}^{t}$

$$\mathcal{L}_{\text{CLIP}}^{t} = \min\left\{ \frac{\pi\left(l^{t+1} \middle| e^{t}, h_{\pi}^{t}\right)}{\pi_{\text{old}}\left(l^{t+1} \middle| e^{t}, h_{\pi}^{t}\right)} \hat{f}_{\text{ad}}^{t} \right. \left. \text{CLIP}\left(\frac{\pi\left(l^{t+1} \middle| e^{t}, h_{\pi}^{t}\right)}{\pi_{\text{old}}\left(l^{t+1} \middle| e^{t}, h_{\pi}^{t}\right)} \right) \hat{f}_{\text{ad}}^{t} \right\} \tag{18}$$

where CLIP is the operation that clips input to $[0.8, 1.2]$ in our experiments.

Then, we use the following loss function to boost the exploration of the optimal policy network π and the precise estimated advantages by V

$$\max_{\pi, V} \mathbb{E}_{x,t}\left[L_{\text{CLIP}}^{t} - \lambda_1 \text{MSE}\left(V\left(e^{t}, h_{\pi}^{t}, \sum_{t \le i \le k} \gamma^{i-t} R^{t} \right) \right) + \lambda_2 S_{\pi}\left(e_{t}, h_{\pi}^{t}\right) \right] \tag{19}$$

where $\lambda_1 = 0.5$ and $\lambda_2 = 0.01$ are two parameters to smooth the loss function; MSE denotes the mean square error loss function; and $S_{\pi}(e^{t}, h_{\pi}^{t})$ denotes the entropy bonus for the current actor–critic network following [34], [49].

F. Semantic Analysis Calculation

Given an instance x, it has k attribute groups $\{g^1, g^2, \ldots, g^k\}$. From datasets, we have semantic group annotated by humans $\{g_s^1, g_s^2, \ldots\}$. For an attribute group g^i and an semantic group annotated by humans g_s^j, we can calculate the semantic ratios of different semantic groups in g^i by

$$o_{g_s^j}^{g^i} = \frac{\left| g^i \cap g_s^j \right|}{\left| g^i \right|} \tag{20}$$

where $o_{g_s^j}^{g^i}$ denotes the ratio of semantic group g_s^j in attribute group g^i; $|g^i \cap g_s^j|$ denotes the attribute number of the overlapping attributes in g^i and g_s^j; and $|g^i|$ denotes attribute number in g^i. Apparently, $\sum_j o_{g_s^j}^{g^i} = 1$.

To ease the imbalance of criterion numbers in different manual semantic groups, we normalize the ratios to know the relative ratios of semantic groups for each attribute group by

$$\text{no}_{g_s^j}^{g^i} = \frac{o_{g_s^j}^{g^i}}{\max\left(\left\{ \left\| o_{g_s^j}^{g^i} \right\|_2 : i \in [1, k] \right\} \right)} \tag{21}$$

where $\|o_{g_s^j}^{g^i}\|_2$ denotes the L2-norm of $o_{g_s^j}^{g^i}$; $\text{no}_{g_s^j}^{g^i}$ denotes the normalized ratio of the semantic group g_s^j in attribute group g^i.

To better analyze the ratios, we rescale the ratio scope and let their sum be 1 by Softmax

$$\text{ro}_{g_s^j}^{g^i} = \frac{\exp\left(\text{no}_{g_s^j}^{g^i}\right)}{\sum_j \exp\left(\text{no}_{g_s^j}^{g^i}\right)} \tag{22}$$

where $\text{ro}_{g_s^j}^{g^i}$ denotes the relative ratio of semantic group g_s^j in the attribute group g^i; $\sum_j \text{ro}_{g_s^j}^{g^i} = 1$.

Then, let $\text{ro}_{g_s^j}^{g^i}(x)$ denote the ratio of the semantic group g_s^j in the attribute group g^i for instance x. We can know the average relative ratios of semantic groups in attribute groups on each dataset, which can reveal the semantic tendencies

$$\text{do}_{g_s^j}^{g^i} = \frac{\sum_{x \in X} \text{ro}_{g_s^j}^{g^i}(x)}{|X|} \tag{23}$$

where X denotes a dataset and $|X|$ denotes the instance number.

We use $\text{do}_{g_s^j}^{g^i}$ to portray the semantic tendencies of the learned attribute groups. Only if g^i has a high relative ratio for each instance in a dataset, we can $\text{do}_{g_s^j}^{g^i}$ achieve a high score, which means that g^i focuses on learning g_s^j. Otherwise, it does not focus on g_s^j. Therefore, $\text{do}_{g_s^j}^{g^i}$ can represent the semantic tendencies of the attribute groups.

REFERENCES

[1] C. H. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 951–958.

[2] V. K. Verma and P. Rai, "A simple exponential family framework for zero-shot learning," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*. Skopje, Macedonia: Springer, 2017, pp. 792–808.

[3] J. Li, M. Jing, K. Lu, Z. Ding, L. Zhu, and Z. Huang, "Leveraging the invariant side of generative zero-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7402–7411.

[4] W. Xu, Y. Xian, J. Wang, B. Schiele, and Z. Akata, "Attribute prototype network for zero-shot learning," in *Proc. 34th Conf. Neural Inf. Process. Syst.* Red Hook, NY, USA: Curran Associates, 2020, pp. 21969–21980.

[5] G.-S. Xie *et al.*, "Attentive region embedding network for zero-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9384–9393.

[6] L. Chen, H. Zhang, J. Xiao, W. Liu, and S.-F. Chang, "Zero-shot visual recognition using semantics-preserving adversarial embedding networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1043–1052.

[7] R. Felix, V. B. Kumar, I. Reid, and G. Carneiro, "Multi-modal cycle-consistent generalized zero-shot learning," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 21–37.

[8] Y. Xian, T. Lorenz, B. Schiele, and Z. Akata, "Feature generating networks for zero-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5542–5551.

[9] J. Song, C. Shen, Y. Yang, Y. Liu, and M. Song, "Transductive unbiased embedding for zero-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1024–1033.

[10] G.-S. Xie *et al.*, "Region graph embedding network for zero-shot learning," in *Proc. Eur. Conf. Comput. Vis.* Glasgow, U.K.: Springer, 2020, pp. 562–580.

[11] Y. Atzmon and G. Chechik, "Probabilistic AND-OR attribute grouping for zero-shot learning," 2018, *arXiv:1806.02664*.

[12] D. Jayaraman, F. Sha, and K. Grauman, "Decorrelating semantic visual attributes by resisting the urge to share," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1629–1636.

[13] X. Wang, Q. Li, P. Gong, and Y. Cheng, "Zero-shot learning based on multitask extended attribute groups," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 51, no. 3, pp. 2003–2011, Mar. 2021.

[14] Y. Long and L. Shao, "Describing unseen classes by exemplars: Zero-shot learning using grouped simile ensemble," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2017, pp. 907–915.

[15] C. S. Coelho and D. R. Moles, "Student perceptions of a spiral curriculum," *Eur. J. Dental Educ.*, vol. 20, no. 3, pp. 161–166, Aug. 2016.

[16] W. M. Clark, D. DiBiasio, and A. G. Dixon, "A project-based, spiral curriculum for introductory courses in ChE: Part 1. Curriculum design," *Chem. Eng. Educ.*, vol. 34, no. 3, pp. 222–233, 2000.

[17] J. S. Bruner, *The Process of Education*. Cambridge, MA, USA: Harvard Univ. Press, 2009.

[18] Y. Yun, S. Wang, M. Hou, and Q. Gao, "Attributes learning network for generalized zero-shot learning," *Neural Netw.*, vol. 150, pp. 112–118, Jun. 2022.

[19] Y. Yang, L. Zhao, and X. Liu, "Iterative zero-shot localization via semantic-assisted location network," *IEEE Robot. Autom. Lett.*, vol. 7, no. 3, pp. 5974–5981, Jul. 2022.

[20] M. Ye and Y. Guo, "Progressive ensemble networks for zero-shot recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 11728–11736.

[21] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. S. Torr, and T. M. Hospedales, "Learning to compare: Relation network for few-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1199–1208.

[22] L. Zhang, T. Xiang, and S. Gong, "Learning a deep embedding model for zero-shot learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2021–2030.

[23] Y. Shigeto, I. Suzuki, K. Hara, M. Shimbo, and Y. Matsumoto, "Ridge regression, hubness, and zero-shot learning," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*. Porto, Portugal: Springer, 2015, pp. 135–151.

[24] Y. Zhu, J. Xie, Z. Tang, X. Peng, and A. Elgammal, "Semantic-guided multi-attention localization for zero-shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 14943–14953.

[25] S. Yang, K. Wang, L. Herranz, and J. van de Weijer, "Simple and effective localized attribute representations for zero-shot learning," 2020, *arXiv:2006.05938*.

[26] Y. Liu, J. Guo, D. Cai, and X. He, "Attribute attention for semantic disambiguation in zero-shot learning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6698–6707.

[27] S. Mathe, A. Pirinen, and C. Sminchisescu, "Reinforcement learning for visual object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2894–2902.

[28] A. Pirinen and C. Sminchisescu, "Deep reinforcement learning of region proposal networks for object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6945–6954.

[29] Y. Wang, K. Lv, R. Huang, S. Song, L. Yang, and G. Huang, "Glance and focus: A dynamic approach to reducing spatial redundancy in image classification," in *Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 2432–2444.

[30] T. Chen, Z. Wang, G. Li, and L. Lin, "Recurrent attentional reinforcement learning for multi-label image recognition," in *Proc. AAAI Conf. Artif. Intell.*, 2018, vol. 32, no. 1, pp. 6730–6737.

[31] W.-H. Chu, Y.-J. Li, J.-C. Chang, and Y.-C.-F. Wang, "Spot and learn: A maximum-entropy patch sampler for few-shot image classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6251–6260.

[32] O. Dogru, K. Velswamy, and B. Huang, "Actor–critic reinforcement learning and application in developing computer-vision-based interface tracking," *Engineering*, vol. 7, no. 9, pp. 1248–1261, 2021.

[33] J. Singh and L. Zheng, "Combining semantic guidance and deep reinforcement learning for generating human level paintings," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 16387–16396.

[34] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017, *arXiv:1707.06347*.

[35] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 2672–2680.

[36] W.-L. Chao, S. Changpinyo, B. Gong, and F. Sha, "An empirical study and analysis of generalized zero-shot learning for object recognition in the wild," in *Proc. Eur. Conf. Comput. Vis.* Amsterdam, The Netherlands: Springer, 2016, pp. 52–68.

[37] G. Patterson and J. Hays, "SUN attribute database: Discovering, annotating, and recognizing scene attributes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2751–2758.

[38] P. Welinder *et al.*, "Caltech-UCSD birds 200," Comput. Neural Syst., California Inst. Technol., Pasadena, CA, USA, Tech. Rep. 2010-001, 2010.

[39] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth, "Describing objects by their attributes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 1778–1785.

[40] Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata, "Zero-shot learning—A comprehensive evaluation of the good, the bad and the ugly," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 9, pp. 2251–2265, Sep. 2019.

[41] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proc. COMPSTAT'*. Paris, France: Springer, 2010, pp. 177–186.

[42] S. Biswas and Y. Annadani, "Preserving semantic relations for zero-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7603–7612.

[43] H. Zhang, Y. Long, Y. Guan, and L. Shao, "Triple verification network for generalized zero-shot learning," *IEEE Trans. Image Process.*, vol. 28, no. 1, pp. 506–517, Jan. 2019.

[44] R. Gao *et al.*, "Zero-VAE-GAN: Generating unseen features for generalized and transductive zero-shot learning," *IEEE Trans. Image Process.*, vol. 29, pp. 3665–3680, 2020.

[45] M. K. Yucel, R. G. Cinbis, and P. Duygulu, "A deep dive into adversarial robustness in zero-shot learning," in *Proc. Eur. Conf. Comput. Vis.* Glasgow, U.K.: Springer, 2020, pp. 3–21.

[46] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[47] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.

[48] P. Dangeti, *Statistics for Machine Learning*. Birmingham, U.K.: Packt, 2017.

[49] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Mach. Learn.*, vol. 8, nos. 3–4, pp. 229–256, 1992.

Zhe Liu received the Ph.D. degree from the School of Computer Science and Engineering, University of New South Wales (UNSW), Sydney, NSW, Australia.

He is currently a Lecturer with the Jiangsu Provincial Engineering Laboratory of Pattern Recognition and Computational Intelligence, Jiangnan University, Wuxi, China. His current research interests include zero-shot learning, generative networks, data mining and their applications to computer vision, and human activity recognition.

Yun Li is currently pursuing the Ph.D. degree with the School of Computer Science and Engineering, University of New South Wales (UNSW), Sydney, NSW, Australia.

Her current research interests include zero-shot learning, attentive networks, machine learning and their applications to computer vision, and genetic engineering.

Lina Yao (Senior Member, IEEE) is currently a Scientia Associate Professor with the School of Computer Science and Engineering, University of New South Wales (UNSW), Sydney, NSW, Australia. Her current research interests include data mining and machine learning with applications to the Internet of Things, information filtering and recommending, human activity recognition, and brain–computer interface.

Julian McAuley received the Ph.D. degree from The Australian National University, Canberra, ACT, Australia, in 2011.

He has been a Professor with the Computer Science Department, University of California, San Diego, La Jolla, CA, USA, since 2014, where he works on applications of machine learning to problems involving personalization and teaches classes on the personalized recommendation. He likes bicycling and baroque keyboard. Previously, he was a Post-Doctoral Scholar with Stanford University, Stanford, CA, USA. His research is concerned with developing predictive models of human behavior using large volumes of online activity data.

Sam Dixon received the bachelor's degree (Hons.) in computer science from The University of Western Australia, Perth, WA, Australia, in 2020, researching information extraction from cyber-security threat reports. He is currently pursuing the Ph.D. degree in computer science with the University of New South Wales, Sydney, NSW, Australia.

His research lies in developing generalizable, multimodal machine learning algorithms.