Simple Primitives With Feasibility- and Contextuality-Dependence for Open-World Compositional Zero-Shot Learning

Zhe Liu, Yun Li, Lina Yao, Senior Member, IEEE, Xiaojun Chang, Senior Member, IEEE, Wei Fang, Member, IEEE, Xiaojun Wu, Member, IEEE, and Abdulmotaleb El Saddik, Fellow, IEEE

Abstract—The task of Open-World Compositional Zero-Shot Learning (OW-CZSL) is to recognize novel state-object compositions in images from all possible compositions, where the novel compositions are absent during the training stage. The performance of conventional methods degrades significantly due to the large cardinality of possible compositions. Some recent works consider simple primitives (i.e., states and objects) independent and separately predict them to reduce cardinality. However, it ignores the heavy dependence between states, objects, and compositions. In this paper, we model the dependence via feasibility and contextuality. Feasibility-dependence refers to the unequal feasibility of compositions, e.g., hairy is more feasible with cat than with building in the real world. Contextuality-dependence represents the contextual variance in images, e.g., cat shows diverse appearances when it is dry or wet. We design Semantic Attention (SA) to capture the feasibility semantics to alleviate impossible predictions, driven by the visual similarity between simple primitives. We also propose a Generative Knowledge Disentanglement (KD) to disentangle images into unbiased representations, easing the contextual bias. Moreover, we complement the independent compositional probability model with the learned feasibility and contextuality compatibly. In the experiments, we demonstrate our superior or competitive performance, SA-and-KD-guided Simple Primitives (SAD-SP), on three benchmark datasets.

Index Terms—Attention network, compositional zero-shot learning, generative network, knowledge disentanglement, open world.

I. INTRODUCTION

Many datasets exhibit long-tailed distribution, i.e., a large number of classes have few or even no prior instances [1], [2], [3], [4], [5], [6], [7], [8]. Insufficient data become a bottleneck limiting the universality of deep learning [9], [10], [11], [12], [13], [14], [15]. Comparatively, humans can intuitively identify non-existent concepts (e.g., canvas tree), once humans understand the underlying primitives (e.g., canvas and tree).

Inspired by this, recent works [16], [17], [18], [19], [20] propose a new learning paradigm named Compositional Zero-Shot Learning (CZSL). CZSL models images as compositions of primitive state and object concepts [21], [22], [23], [24]. It aims to extract states and objects in seen images, transferring knowledge from seen to unseen, thereby recognizing unseen state-object compositions without training. For example, given images of canvas shoe and brown tree, machines can learn simple primitives of shoe and brown, thus directly recognizing the unseen composition of brown shoe in images.

CZSL is challenging due to the context-dependent appearances [17], [18]. For example, small scales differently for elephants and cats; cats look different when they are young or old. In other words, states and objects lead to visual changes to each other. Simple primitives in images are entangled visually [25]. Most current works [16], [19], [25], [26], [27], [28] view states or objects as the bases. They simulate the visual changes on bases caused by contextuality, inferring the possible entangled embedding. Thus, they can learn the visual embedding specific to each composition, helping distinguish unseen images. Despite the good performance of this strategy in Closed-World CZSL (CW-CZSL), it tends to largely degrade in Open-World CZSL (OW-CZSL) [21], [22], [29]. CW-CZSL provides the possible compositions as priors to simplify the inference, while OW-CZSL anonymizes these compositions and predicts compositions in the entire compositional space. For example, C-GQA [29] (one of the benchmark datasets) contains 431 states and 674 objects. CW-CZSL performs inferences on a limited search space of 6,515 compositions (5,592 seen compositions for training and 923 unseen compositions for testing).
approximately 2% of the entire compositional space (278,362 possible compositions) in OW-CZSL. The large cardinality of possible compositions largely impairs the discrimination ability of compositional embedding [21].

Some OW-CZSL methods [19], [21], [22], [29] exploit the unequal feasibility to ease impairment of the large learning cardinality. These methods assign each composition a feasibility score to represent the probability of its existence, and they discard the impossible compositions. For example, canvas tree is much less likely to exist in the world than canvas shoe. Models can assign a large feasibility to canvas shoe but a small feasibility to canvas tree, eliminating canvas tree by a feasibility threshold to make models focus on real-world images. Current methods usually calculate the feasibility relying on external semantic knowledge [21] or pairwise comparisons of label embedding [19], [22], [29]. It may be infeasible when datasets lack relevant semantic information or have a large number of simple primitives. Some other works [18], [21], [23] decompose the composition into simple primitives. They assume that states/objects follow independent probability distributions. The compositional probability distribution can be the joint distribution of simple primitives, i.e., 
\[ p(\text{composition}|\text{image}) = p(\text{state}|\text{image})p(\text{object}|\text{image}). \]
Since the cardinality of the simple primitives is much smaller than the cardinality of the compositions, the discrimination ability can be better preserved. However, the assumption of independent distributions ignores contextual and feasible relations between simple primitives [21], leading to biased predictions.

Motivated by the above issues, we introduce two ideas to enable independent simple primitives to be compatible with learning contextuality and feasibility. First, inspired by the finding that the feasibility of similar states is shared among similar objects [29], we assume that similar visual primitives share similar compositional feasibility. As shown in Fig. 1 (left), striped and cat are two simple primitives learned from the image independently. Similar visual patterns (e.g., striped and spotted) and species (e.g., cat and tiger) tend to be exchangeable in the compositions. Therefore, we may exchange simple primitives in seen compositions to compose some new possible compositions, e.g., spotted cat and dry tiger, which tend to have higher feasibility than other unseen compositions. Taking advantage of the point that all objects and states are seen in OW-CZSL, we can utilize the semantics in existing compositions to infer the feasibility distribution of all possible compositions. Second, since it may be infeasible to simulate all visual changes of contextuality on states/objects, we tackle contextuality by disentangling the entangled visuals into disentangled feature representations. In Fig. 1 (right), striped is not identical on different species, e.g., cat and hyena. Models might classify striped patterns into different classes due to the difference. We propose to disentangle images into unbiased feature representations that do not contain visual changes of contextuality. For example, we learn disentangled representations of striped not containing information of cat or hyena. Then, striped patterns on different species can better be recognized as a unified class. Similarly, we can learn a unified cat under different states.

In this paper, we implement our ideas in a unified model called Semantic Attention and Knowledge Disentanglement guided Simple Primitives (SAD-SP). SAD-SP consists of three branches: Simple Primitive (SP), Semantic Attention (SA), and Knowledge Disentanglement (KD). SP follows the conventional practice of learning independent probability distributions for simple primitives. SA is a simple attention module driven by the inherent visual similarity between primitives to learn the shared compositional feasibility. KD is fulfilled by a generative adversarial network. We propose a distributional loss to supervise the generative network to exclude contextual information for unbiased representations. Both SA and KD have two parallel networks corresponding to the state and object, respectively. Unlike conventional methods taking SP as the final predictions, we use probabilistic revisions of the feasibility and the contextuality to complement the independent probability distributions.

Our contributions can be summarized as follows: 1) We propose two compatible ideas to complement conventional SP learning in a unified model. The unified model combines the learning of feasibility, contextuality, and independent simple primitives in a probabilistic format, which keeps a small learning cardinality. 2) We design the parallel SA and KD to learn state/object-specific information. Our SA can infer compositional feasibility without external knowledge or pairwise similarity comparison. We propose a new distributional loss to enable KD to learn unbiased representation. 3) We conduct extensive experiments to show the outperformance or the competitive performance of SAD-SP on benchmark datasets in the open-world setting. We show detailed quantitative and qualitative analyses to prove the model’s effectiveness in improving compositional predictions.
II. RELATED WORK

A. Compositional Zero-Shot Learning

Compositional learning [25], [30], [31], [32], [33], [34], [35] aims to enable models to learn simple primitives in images. Compositional zero-shot learning further recognizes unseen compositions in images composed of seen simple primitives. Simple primitives usually consist of two types of primitive concepts, e.g., states and objects [18], [26], [36], [37]. CZSL works [16], [17], [18], [22], [23] attempt to learn discriminative visual representations of concepts to conduct classification. These methods can be divided into two main categories: compositional embedding and simple primitive. Methods [16], [18], [19], [38] based on compositional embedding project visual information into a common embedding space to recognize compositions directly. For example, Misra et al. [18] configure a specific compositional classifier for each composition to project inputs into the corresponding compositional space for classification. Nagarajan et al. [16] view states as operators and perform all possible transformations on objects to build the embedding space for compositional classification. Contrary to directly classifying compositions, simple primitive methods predict primitive concepts independently and then construct the compositional probability distribution for classification jointly [18], [21], [23], [39]. These methods assume that states/objects follow independent probability distributions. They consider the compositional probability distribution as the joint probability distribution of objects and states, i.e., the product of predicted probabilities of objects and states. In other words, the problem of recognizing compositions is decomposed into recognizing two simple primitives independently. For example, Misra et al. [18] apply two independent classifiers to separately predict simple primitives and take the product of probabilities as the probability of compositions. Karthik et al. [23] further propose a scalar bias to ease biased predictions towards seen compositions and enhance performance. Our method is similar to simple primitive methods but considers simple primitives’ contextuality- and feasibility-dependence. We use contextuality and feasibility as revisions to complement state-object dependence in probabilities of simple primitives. The attention is threshold-free, which uses feasibility to calibrate probabilities directly. Thus, we do not need post-processing to select a threshold by hand engineering. Instead, we require a shared hyperparameter set to be used across datasets, which demands little hand-engineering effort to merge modules.

B. Open-World Compositional Zero-Shot Learning

Open-world compositional zero-shot learning is more realistic and challenging than conventional closed-world setting. In the open-world setting, no priors about possible unseen compositions are given. CW-CZSL works tend to be less effective due to the large cardinality of possible compositions [16], [17], [18], [20], [21]. However, some works in CW-CZSL can still be enlightening to tackle the large cardinality. For example, for example, Khan et al. [40] utilize the self-attention mechanism to learn interdependency structure between compositions via identifying propagating routes for better label embedding. Li et al. [19] mask impossible compositions by computing pair probability based on the distance between state and object categories. Xu et al. [41] adopt a key-query-based attention mechanism to capture the correlation between primitive concepts in a graph to pass messages selectively. Atzmon et al. [20] adopt a causal perspective to disentangle images into object and state representations, and consequently predict compositions based on the likelihood of images conditioned on each composition. In OW-CZSL, similar to [19], Mancini et al. [22] propose to utilize the graph structure to model the dependence between state, object, and compositions. In this way, some infeasible compositions can be eliminated during inference. However, these feasibility computation is based on pairwise similarity comparison. Pairwise comparison tends to be less effective on datasets with large numbers of states and objects [21]. Another way to reduce composition cardinality is simple primitives [18], [21], [23], [42]. The paradigm of simple primitives independently predicting states and objects can naturally reduce cardinality. Karthik et al. [21] further improve simple primitives by estimating the feasibility of each composition through external knowledge to eliminate impossible state-object pairs. Different from [21], [41], our work estimates feasibility with attention mechanism and implies feasibility in the form of probability to enhance simple primitive learning.

C. Knowledge Disentanglement in Zero-Shot Learning

Visual variance in images caused by contextuality is a common issue in zero-shot learning [43], [44], [45], [46], [47], [48], [49], [50]. Plenty of previous works have been done to utilize knowledge disentanglement to learn invariant or unbiased visual representations. For example, Chen et al. [49] and Li et al. [50] use a conditional VAE to disentangle images into semantic-consistent and semantic-unrelated latent vectors. Li et al. [51] generatively disentangle inputs into sub-features to obtain simple independent hierarchical features. Knowledge disentanglement can also be applied in disentangling graphs. Geng et al. [52] disentangle knowledge graphs into ontology embeddings to capture fine-grained semantic information. In compositional zero-shot learning, Yang et al. [25] disentangle images into states and objects based on the causal effects in compositions. Saini et al. [28] apply a state/object affinity network to disentangle the same states or objects from image pairs contrastively. Zhang et al. [27] reconsider CZSL as an out-of-distribution generalization problem and use domain alignment in the gradient level to disentangle images into object-invariant and attribute-invariant features. In this paper, we propose a novel generative network to disentangle images based on the probability distributions of classes, which can be effective in the OW-CZSL setting.

III. METHOD

A. Problem Formulation

Given an image dataset \( X = \{x_i : i \in [1, N]\} \) with compositional labels \( Y = \{y_i : i \in [1, N]\} \), \( N \) denotes the dataset size,
and \((x_i, y_i)\) is an image \(x_i\) with the corresponding compositional label \(y_i\). CZSL [29] aims to recognize state-object compositions \(y_i = (s_i, o_i)\) based on the given sets of states \(S\) and objects \(O\). In this paper, we focus on the challenging OW-CZSL. OW-CZSL recognizes both seen and unseen object-state compositions from the whole compositional space, i.e., \(Y_{\text{space}} = S \times O\). OW-CZSL divides \(X\) into a training set \((X_{\text{tr}}, Y_{\text{tr}})\), a validation set \((X_{\text{val}}, Y_{\text{val}})\) and a test set \((X_{\text{ts}}, Y_{\text{ts}})\). The training set is used to train the model, where \(Y_{\text{tr}}\) contains all possible states and objects but not all possible compositions. The validation set and the test set consist of both seen and unseen compositions during the training stage. Note that seen compositions in the training set have no intersection with unseen compositions in the validation set or the test set.

B. Simple Primitives in CZSL

Following the SP baselines in OW-CZSL [18], [21], we predict the probabilities of states/objects independently and take the product of their probabilities as the compositional probabilities to recognize compositions. Our SP consists of two parts: extractor and classifier. The extractor learns the state/object feature representations of the input. The classifier predicts the state/object probabilities based on the learned feature representations.

Formally, given an image \(x\), we have an extractor \(f_e: X \rightarrow Z\) embedding the image \(x\) to feature representation \(z\). Note that \(f_e\) utilizes a shared backbone, along with two additional independent FCNs for learning state and object representations, denoted as \(Z = \{Z_s, Z_o\}\). We regard \(z^s \in Z_s\) and \(z^o \in Z_o\) as the corresponding visual information learned for subsequent predictions or analyses. Then, a state classifier \(f_s: Z_s \rightarrow \Delta_S\) and an object classifier \(f_o: Z_o \rightarrow \Delta_O\) predicts the state probability \(p_s = [p_{s_1}, \ldots, p_{s_N}]\) and object probability \(p_o = [p_{o_1}, \ldots, p_{o_M}]\), respectively, where \(|S|\) and \(|O|\) denote the number of states and objects in the dataset. \(p_s\) and \(p_o\) are vectors that \(\sum_{k=1}^{N} p_{s_k} = 1\). \(p_{s_k}\) and \(p_{o_j}\) represent the prediction probability of the \(k\)th state and the \(j\)th object, respectively. Then, the compositional probability for each state-object pair can be calculated by the probabilities of independent predictions \(p(s_k, o_j) = p_{s_k} \times p_{o_j}\).

We use the cross-entropy loss to train the predictions of states and objects as follows:

\[
\min_{f_s, f_o, f_{e}} \mathcal{L}_{\text{sp}} = \sum_{i=1}^{N} \mathcal{L}_{\text{ce}}(f_s(f_e(x_i)), s_i) + \mathcal{L}_{\text{ce}}(f_o(f_e(x_i)), o_i)
= - \sum_{i=1}^{N} \log f_s(z^s_i, s_i) + \log f_o(z^o_i, o_i)
\]

where \(\mathcal{L}_{\text{ce}}\) denotes the cross-entropy loss; \(z_i = f_e(x_i);\) \(z^s_i\) and \(z^o_i\) denote the corresponding branch for predictions in \(z_i;\) \(s_i\) and \(o_i\) are the corresponding ground-truth state and object labels for \(x_i;\) \(f_s(z^s_i, s_i) = p_{s_k}\) is the probability assigned to state \(s_k\) of \(x_i;\) \(f_o(z^o_i, o_i) = p_{o_j}\) is the probability assigned to object \(o_j\) of \(x_i;\)

C. Simple Primitives With Feasibility and Contextuality

In this section, we introduce the details of our proposed semantic attention and knowledge disentanglement to tackle feasible and contextual dependence between simple primitives. The model overview of SAD-SP is shown in Fig. 2.

1) Simple Primitives With Dependence: The conventional SP ignores the dependence between states, objects, and compositions, while the dependence can be effective in compositional recognition [22], [38]. Thus, we assume conventional SP as a basis to represent the predictions based on the superficial semantics in images. We let the dependence between simple primitives be the high-level semantic information, influencing the basic probability of superficial semantics. Considering that even images from the same compositions may exhibit different feasible and contextual dependence due to the varying visual patterns in images, we model the dependence on the instance level.

For a given instance and an arbitrary composition \((s_k, o_j)\), we propose the revised composition probability as follows:

\[
p(s_k, o_j) = p'_s \times p'_{o_j}
= (p_{s_k} + p_f(s_k|o_j) + p_c(s_k)) \times (p_{o_j} + p_f(o_j|s_k) + p_c(o_j))
\]

where \(p'_s\) and \(p'_{o_j}\) denote the probability of simple primitives with dependence; \(p_{s_k}\) and \(p_{o_j}\) denote the independent state and object probabilities without dependence; \(p_f\) and \(p_c\) represent the probabilistic revisions of feasibility and contectuality dependent on the instance, respectively.

We argue that each word has its own semantic space. The feasibility of compositions may differ for the state and the object. For example, the feasibility of white polar bear differs for white and polar bear: when we describe polar bears, we may use the state of white frequently; however, when we use white, we may frequently associate it with common white things in daily life, e.g., snow, but not the uncommon polar bears. Furthermore, due to varying degrees of whiteness in images, each polar bear may exhibit different correlations with the color white. For example, whiteness may differ among polar bears in different instances. When in insufficient lighting conditions, the fur of polar bears can be distinctly characterized as white. However, under insufficient illumination, e.g., during sunset, the fur may exhibit shadows, leading to an instance-level whiteness deviation from its inherent whiteness. Thus, directly predicting primitives may ignore such feasibility differences. We use \(p_f(s_k|o_j)\) and \(p_f(o_j|s_k)\) to represent the instance-specific conditioned feasibility of \((s_k, o_j)\) for the state \(s_k\) and the object \(o_j\), respectively.

2) Semantic Attention for Feasibility: We learn state/object-conditioned compositional feasibility via semantic attention following the idea: unseen compositions composed by exchanging similar simple primitives in seen compositions are more feasible than other unseen compositions. We design an attention module to implement our idea, driven by the inherent similarity of feature representation during training: similar visual patterns tend to be embedded as similar feature representations [17]. In other words, similar simple primitives tend to have similar feature
representations. Thus, we can use parallel attention to learn conditioned semantic relations for states and objects, respectively. For example, given compositions wet tiger and small cat, if we use attention to learn from objects to help predict states, the attention will build strong feasibility from tiger to wet and from cat to small. Due to the similarity between tiger and cat, when the object is tiger or cat, the attention tends to assign high feasibility to state wet and state small, which is consistent with our idea.

Following previous work on end-to-end training [53], [54], [55], we view the second-last layer output of SP as the instance-specific indicator of SP predictions, which can be used as $Z$. Then, we can learn an object-conditioned attention for state $f_{sa}: Z_o \rightarrow A_s$ and a state-conditioned attention for object $f_{oa}: Z_s \rightarrow A_o$. We use the Sigmoid function to let $a^s_i \in (0, 1)^{|S|}$ and $a^o_i \in (0, 1)^{|O|}$ be vectors that have the same size as the state/object set. Each element in the vector represents the strength of feasibility between the corresponding state and object. The larger element means the larger probability of the composition of the state and object existing in the dataset. We use the learned semantics to improve the search space of SP by fusing the attention map with the predictions via the element-wise product, i.e., $p_f(s_k|o_j) = a^s_i \times p_{sk}, p_f(o_j|s_k) = a^o_j \times p_{o_j}$. Then, we minimize the following loss for training:

$$L_{att} = \sum_{i=1}^{N} L_{ce}(f_{sa}(z^o_i) \otimes f_s(z^s_i, s_i)) + L_{ce}(f_{oa}(z^s_i) \otimes f_o(z^o_i, o_i)) = -\sum_{i=1}^{N} \log a^s_{\sigma(s_i)} \otimes f_s(z^s_i, s_i) + \log a^o_{\sigma(o_i)} \otimes f_o(z^o_i, o_i)$$

(3)

where $\sigma(s_i)$ and $\sigma(o_i)$ return the location of the ground-truth labels in attention, respectively; $a^s_{\sigma(s_i)} \otimes f_s(z^s_i, s_i) = a^o_{\sigma(o_i)} \otimes f_o(z^o_i, o_i)$ denote the state and object branch of $z_i^s$ and $z_i^o$ respectively. The generators are trained in an adversarial way with a pair of parallel classifiers and discriminators. Specifically, $<f_{ds}, f_{do}>$ is for state/object predictions $L_{dc}$; $<f_{sd}, f_{ado}>$ is for denosing state/object information $L_{dis}$; $<f_{ds}, f_{do}>$ is for distinguishing real/fake features $L_{dis}; L_{dis}$ is the overall loss for KD.

$L_{att}$ trains attention maps $a^s$ and $a^o$ to learn complementary feasibility probabilities for enhancing the final predictions. Specifically, the attention maps learn semantic relations from existing images to measure the composition probabilities dependent on states and objects. The additional original scores $L_{ce}$ from the SP module can be viewed as a regularization term, which prevents the overfitting of revised predictions on seen compositions and enables the SA module to learn to enhance SP. $L_{att}$ shifts the model focuses to possible compositions to refine the search space. Consequently, the probabilistic revision of objects/states can be considered as the auxiliary classification information learned from the states/objects to help enhance the original predictions, which are guided by the semantic relations between simple primitives. Since we view the feasibility as auxiliary information to make predictions focused on possible compositions, we do not need a post-processing threshold to help eliminate the impossible compositions [21].

3) Knowledge Disentanglement for Contextualiy: Similar to SA, we take the second-last layer output of SP as the entangled feature representations $Z = \{Z_s, Z_o\}$. We disentangle feature representations to obtain instance-specific unbiased visual information in an adversarial way. We apply two parallel generators, discriminators, and classifiers to disentangle the state/object. We use a state generator $f_{sg}: Z_s \rightarrow Z'_s$ and an object generator $f_{og}: Z_o \rightarrow Z'_o$ to generate the disentangled state representation and object representation, respectively. We propose two principles to ensure the disentangled representations of states and objects:

(I) The disentangled feature representations should accurately
predict the target classes while disregarding non-target classes. For example, the disentangled state representations should only retain the state information without any object information. (II) The disentangled feature representations should be as ‘real’ as the originally learned feature representations. The word ‘real’ refers to that the disentangled representations should carry as much essential state/object information as the original representations. Principle (I) ensures that the model learns unbiased feature representations, while principle (II) serves as a regularization requirement that prevents eliminating too much meaningful visual information during the disentanglement.

To fulfill principle (I), we design disentangled classifiers and denoising classifiers to supervise the model learning required information. The disentangled state classifier \( f_{ds}: \{Z_s, Z_s'\} \rightarrow \Delta_S \) and the disentangled object classifier \( f_{do}: \{Z_o, Z_o'\} \rightarrow \Delta_O \) supervise the disentangled feature representations (i.e., \( z^*_s \) and \( z^*_o \)) carrying the same classification information as the original feature representations (i.e., \( z^*_s \) and \( z^*_o \)). We use denoising classifiers \( f_{s,den} : \{Z_s, Z_s'\} \rightarrow \{1_S, U_S\} \) and \( f_{o,den} : \{Z_o, Z_o'\} \rightarrow \{1_O, U_O\} \) to disentangle and denoise the non-target information in the disentangled feature representations, where \( U \) denotes the uniform distribution and 1 denotes the one-hot distribution. We use the uniform distribution, i.e., \( U_S \) and \( U_O \), to represent ‘disentangled’, which means that the feature representations cannot distinguish classes. All the classes have the same probability of \( \frac{1}{N} \) or \( \frac{1}{|S|, |O|} \) in predictions. We let the one-hot distributions, i.e., \( 1_S \) and \( 1_O \), represent ‘non-disentangled’, which means that the feature representations still can accurately assign a probability of 1 to the ground-truth class of state or object. Then, we can use cross-entropy loss and mean square error loss to learn disentangled features as follows:

\[
\begin{align}
\min_{f_{ds}, f_{do}} \mathcal{L}_{dc} &= -\mathbb{E}_{z_i \sim Z} \sum_{i=1}^{N} \log f_{ds}(z^*_s, s_i) + \log f_{do}(z^*_o, o_i) \\
\max_{f_{s,den}} \mathcal{L}_{den} &= -\mathbb{E}_{z_i \sim Z} \sum_{i=1}^{N} \| f_{o,den}(z^*_o, o_i) - 1_O(z_i) \|^2 \\
&\quad + \mathbb{E}_{z_i \sim Z} \sum_{i=1}^{N} \| f_{s,den}(z^*_s, s_i) - 1_S(z_i) \|^2 \\
\min_{f_{g}} \mathcal{L}_{den} &= \mathbb{E}_{z_i \sim Z} \sum_{i=1}^{N} \| f_{o,den}(z^*_o, o_i) - U_O(z_i) \|^2 \\
&\quad + \mathbb{E}_{z_i \sim Z} \sum_{i=1}^{N} \| f_{s,den}(z^*_s, s_i) - U_S(z_i) \|^2
\end{align}
\]

where \( Z' = \{ Z'_s, Z'_o \} \); \( f_d = \{ f_{ds}, f_{do} \} \) denotes the set of disentangled classifiers; \( f_g = \{ f_{sg}, f_{og} \} \) denotes the generators to disentangle knowledge; \( f_{s,den} = \{ f_{s,den}, f_{o,den} \} \) denotes the classifiers to denoise non-target knowledge, \( f_{ds}(z^*_s, s_i) \) and \( f_{do}(z^*_o, o_i) \) denote class probability of \( f_{ds,d} \) assigned to the ground-truth label \( s_i/o_i \) of the input, \( f_{o,den}(z^*_o, o_i) \) and \( f_{s,den}(z^*_s, s_i) \) denote the predicted state/object probability distribution from feature representations of object/state; \( 1_{S/O}(z_i) \) denotes a one-hot distribution that only the ground-truth state/object label of \( z_i \) is one while other elements are zero; \( U_{S/O}(z_i) \) denotes a uniform distribution that each class has an equal probability in the predictions of states/objects for \( z_i \). The disentangled classifier \( f_{ds,d} \) optimizes the disentangled classifier to learn the target classification information from both generated and original feature representation. Then, \( \mathcal{L}_{dc} \) supervises the generator to generate features with the classification information of the target state/object classes.

\( \mathcal{L}_{den} \) is a 2-stage min-max loss function. \( \mathcal{L}_{den} \) first optimizes \( \{ f_{s,den}, f_{o,den} \} \) to be capable of precisely predicting non-target classes. Then, \( \mathcal{L}_{den} \) supervises generators to disentangle feature representations, updating generators to denoise the non-target information in feature representations, i.e., denoising object/state information in \( z^*_s/z^*_o \).

To fulfill principle (II), we use discriminators \( f_{s,dis} : \{Z_s, Z'_s\} \rightarrow \{0, 1\} \) and \( f_{o,dis} : \{Z_o, Z'_o\} \rightarrow \{0, 1\} \) to distinguish real and fake feature representations of states and objects, where 0 denotes ‘fake’ and 1 denotes ‘real’. We consider the generated feature representations as ‘fake’ and the original feature representations as ‘real’. The objective of discriminators is to deceive the discriminators, thereby retaining more visual information from the original inputs. We optimize the generator and discriminator with cross-entropy loss in a min-max format as follows:

\[
\begin{align}
\max_{f_{dis}} \mathcal{L}_{dis} &= \mathbb{E}_{z_i \sim Z} \left[ \log f_{s,dis}(z^*_s, s_i) + \log f_{o,dis}(z^*_o, o_i) \right] \\
&\quad + \mathbb{E}_{z_i \sim Z} \left[ \log (1 - f_{s,dis}(z^*_s, s_i)) + \log (1 - f_{o,dis}(z^*_o, o_i)) \right] \\
\min_{f_{g}} \mathcal{L}_{dis} &= -\mathbb{E}_{z_i \sim Z} \left[ \log f_{s,dis}(z^*_s, s_i) + \log f_{o,dis}(z^*_o, o_i) \right]
\end{align}
\]

where \( f_{dis} = \{ f_{s,dis}, f_{o,dis} \} \) is the discriminators; \( Z' \) represents the generated feature by \( f_{g} \).

\( \mathcal{L}_{dis} \) first optimizes the discriminators to distinguish the generated and original feature representations accurately. It then supervises generators to fool the discriminators, which regularizes the disentangled feature representations to mimic the original representations. This enables KD to focus more on the overall visual patterns and avoid capturing subtle unique patterns. These subtle unique patterns could lead models to build shortcuts from local visual patterns to state/object predictions [54, 58]. Such shortcuts may be biased towards seen compositions, resulting in models lacking the generalization ability to handle both seen and unseen compositions.

Finally, we can summarize the distributional loss \( \mathcal{L}_{kd} \) of knowledge disentanglement in a min-max format as follows:

\[
\begin{align}
\min_{f_{s,gs, f_{o,g}}} \max_{f_{s,dis}, f_{o,dis}} \mathcal{L}_{kd} &= \mathcal{L}_{dc} + \mathcal{L}_{den} + \mathcal{L}_{dis} \\
\end{align}
\]

We can obtain disentangled state and object representations generated by \( f_{sg} \) and \( f_{og} \) for unbiased predictions. We view the unbiased predictions as a probabilistic revision to shift model focus to the predictions without contextual variance, i.e., given the feature representation \( \{ z^*_s, z^*_o \} \) of an image, for an arbitrary
4) Module Combination: The overall loss function of SAD-SP is summarized as follows:

$$\mathcal{L}_{\text{SAD-SP}} = \mathcal{L}_{\text{sp}} + \mathcal{L}_{\text{att}} + \mathcal{L}_{\text{kd}}$$

For an arbitrary image $x_i$, we propose to merge the predictions of modules as follows:

$$\text{argmax}_{(s_k, o_j)} p_{(s_k, o_j)} = p_{s_k}^t \times p_{o_j}^t$$

$$p_{s_k}^t = \gamma_1 f_{s}(z_s^o, s_k) + \gamma_2 a_k^i f_s(z_s^o, s_k) + \gamma_3 f_{ds}(f_{sg}(z_s^o), s_k)$$

$$p_{o_j}^t = \gamma_1 f_{o}(z_o^s, o_j) + \gamma_2 a_j^o f_o(z_o^s, o_j) + \gamma_3 f_{do}(f_{sg}(z_o^s), o_j)$$

where $\gamma_1$, $\gamma_2$, and $\gamma_3$ are hyperparameters tuned on the validation set ($X_{val}, Y_{val}$).

Given that searching for the optimal hyperparameters for each dataset is infeasible, as each hyperparameter has infinitely possible values in the domain of real numbers, we propose to find a set of balanced $\gamma$ as the hyperparameters used across different datasets. Using the same balanced hyperparameters can help us measure the performance improvements brought by our model rather than focusing on careful hyperparameter tuning. To find a set of balanced hyperparameters, we establish three criteria: (I) $\gamma_1 + \gamma_2 + \gamma_3 = 1$; (II) $\gamma_2$ and $\gamma_3$ should range within [0.05, 0.5]; (III) a set of hyperparameters is considered balanced for a dataset if its performance is similar to the performance achieved using the optimal hyperparameters during validation.

Criterion (I) aims to restrict the range of the weighted summation to equal 1, maintaining the weighted summation as a probability. Criterion (II) aims to keep the SP as the main component in the final predictions as we model. Criterion (III) aims to help us determine whether the current hyperparameters are balanced from the perspective of training and inference. During training, $\gamma$ may affect the obtained optimal model parameters; during validation, $\gamma$ may influence the merged results of a trained model. Changing $\gamma$ during training or validation could result in different model performances. Therefore, we consider hyperparameters balanced, if the hyperparameters show good performance during both the training and inference phases.

To fulfill the criteria, we manually select a few sets of hyperparameters meeting the criteria (I-II) and learn the corresponding trained models. We further conduct grid searches for trained models to identify the balanced hyperparameters, measuring the model performance with varying $\gamma_2$ and $\gamma_3$ in the range of [0.05, 0.5] to fulfill criterion (III).

**IV. EXPERIMENT**

**A. Experiment Settings and Implementation Details**

**Datasets:** We evaluate our proposed SAD-SP on three commonly used OW-CZSL benchmark datasets, i.e., UT-Zappos dataset [59, 60], MIT-States dataset [61] and C-GQA dataset [38]. The UT-Zappos dataset is a small dataset only containing shoes. It consists of 12 different shoe types and 16 footwear materials. We view each shoe type as an object class and each footwear material as a state class. Different from UT-Zappos containing only shoes, MIT-States and C-GQA datasets are two large datasets containing diverse objects (e.g., buildings and animals) and states (e.g., shapes and colors). MIT-States dataset has 245 different object classes and 115 state types. C-GQA is a compositional version of Stanford QQA dataset [62] with 674 object categories and 413 state categories. In the open-world scenario, we follow the standard split in previous works [17, 21]: images (~23 k images) from 83 compositions out of 192 possible compositions (~43%) are seen for training on the UT-Zappos dataset; images (~30 k images) from 1,262 compositions out of 28,175 possible compositions (~4%) are used for training on the MIT-States dataset; images (~27 k images) from 5,592 compositions out of 27,8362 possible compositions (~2%) are presented for training on the C-GQA dataset. About 3 k images from 36 compositions (18 seen and 18 unseen compositions), 13 k images from 800 compositions (400 seen and 400 unseen compositions), and 5 k images from 1,811 compositions (888 seen and 923 unseen compositions) are selected for testing on UT-Zappos, MIT-States, and C-GQA, respectively.

**Metrics and Benchmarks:** For a fair comparison, we follow the standard evaluation protocol in previous works [21, 29]. We evaluate the seen accuracy (S), the unseen accuracy (U), the harmonic mean (HM) of seen and unseen accuracy, and the area under the curve (AUC) under the generalized compositional zero-shot setting. In generalized zero-shot learning, models are required to conduct inferences on both seen and unseen classes to show their generalization ability. However, the predictions are often biased to seen classes significantly. We thus use a bias calibration that is commonly used in zero-shot learning [54], [55], [58], [63] and compositional zero-shot learning [17, 21] to ease the biased predictions. Same as the common operation of bias calibration in CZSL [17, 21], we use varying constants as bias terms to measure the best S, best U, best HM, and AUC scores (in %) as our final results.

We compare our methods with 10 representative methods: LE+ [18], AoP [16], TMN [17], SymNet [19], CGE [38], CompCos [29], VisProd [18], VisProd++ [23], KG-SP [21], and Co-CGE [22]. LE+, AoP, TMN, SymNet, CGE, CompCos, and Co-CGE are methods based on the compositional embedding that learns a shared embedding space defined by images and label embeddings to make predictions. VisProd, VisProd++, and KG-SP are similar to our SAD-SP, which takes the probabilistic product of simple primitives to infer the compositions. More specifically, LE+ and VisProd are two baseline models; CGE, CompCos, and Co-CGE enhance LE+ by the feasibility of cosine similarity and/or graph structure between embeddings based on external semantic information, e.g., word2vec+fasttext [64], [65]; VisProd++ and KG-SP improve VisProd by advanced network architecture and bias calibration; AoP, SymNet, and TMN use the generative network or modulator to simulate the contextual variance in images; KG-SP uses external semantic knowledge to inject feasibility into compositional predictions. We take the fixed backbone version of compared methods (e.g., CGE$_{at}$) to evaluate the improvement derived from the end-to-end training.
TABLE I
RESULTS ON THREE OW-CZSL BENCHMARK DATASETS, WHERE FF REPRESENTS THE FIXED BACKBONE; * DENOTES USING EXTERNAL KNOWLEDGE BEIDES IMAGENET

| Model                | MIT-States | UT-Zappos | C-GQA |
|----------------------|------------|-----------|-------|
|                      | S | U | HM | AUC | S | U | HM | AUC | S | U | HM | AUC |
| Compositional Embedding |
| LE+ [18]             | 14.2 | 2.5 | 2.7 | 0.3 | 60.4 | 36.5 | 30.5 | 16.3 | 19.2 | 0.7 | 1.0 | 0.08 |
| AOP [16]             | 16.6 | 5.7 | 4.7 | 0.7 | 50.9 | 34.2 | 29.4 | 13.7 | - | - | - | - |
| TMN [17]             | 12.6 | 0.9 | 1.2 | 0.1 | 55.9 | 18.1 | 21.7 | 8.4 | - | - | - | - |
| SymNet [19]          | 21.4 | 7.0 | 5.8 | 0.8 | 53.3 | 44.6 | 34.5 | 18.5 | 26.7 | 2.2 | 3.3 | 0.43 |
| CompCos* [29]        | 25.3 | 5.5 | 5.9 | 0.9 | 59.8 | 45.6 | 36.3 | 20.8 | 28.0 | 1.0 | 1.6 | 0.20 |
| CompCos* [29]        | 25.4 | 10.0 | 8.9 | 1.6 | 59.3 | 46.8 | 36.9 | 21.3 | 28.4 | 1.8 | 2.8 | 0.39 |
| CDR* [38]            | 29.6 | 4.0 | 4.9 | 0.7 | 58.8 | 46.5 | 38.0 | 21.5 | 28.3 | 1.3 | 2.2 | 0.30 |
| CEG* [38]            | 32.4 | 5.1 | 6.0 | 1.0 | 61.7 | 47.7 | 39.0 | 23.1 | 32.7 | 1.8 | 2.9 | 0.47 |
| Co-CEG* [22]         | 26.4 | 10.4 | 10.1 | 2.0 | 60.1 | 44.3 | 38.1 | 21.3 | 28.7 | 1.6 | 2.6 | 0.37 |
| Co-CEG* [22]         | 30.3 | 11.2 | 10.7 | 2.3 | 61.2 | 45.8 | 40.8 | 23.3 | 32.1 | 3.0 | 4.8 | 0.78 |

The best results are in bold.

in which case the backbone will not be fine-tuned and the input can be viewed as the extracted representations of the backbone.

Notably, we develop an ensemble model, called Ensemble, that incorporates three predictors with identical numbers of FCN layers corresponding to the predictors in modules of SAD-SP. Ensemble regards the prediction summation of three predictors as final predictions, serving as a baseline to discern the improvements contributed by network depth and our proposed methods. We further investigate two variants of SAD-SP, namely GTIPrediction-SAD-SP, to validate the effectiveness of using the second-last layer output of SP as inputs to SA. Both variants use the probability Predictions of SP for states/objects as inputs to SA, but GT-SAD-SP replaces predictions with the one-hot Ground-Truth (GT) labels during training.

Implement Details: We follow the procedure in Section III-C4 and find a set of balanced parameters, i.e., $\gamma$ parameters $\gamma_1 = 0.7$, $\gamma_2 = 0.25$, and $\gamma_3 = 0.05$, based on the validation set of MIT-States. The detailed grid search results can be found in Section IV-C4. The experiments are implemented in PyTorch [69] and NVIDIA TITAN X with CUDA 11.0 [70].

B. Open-World Compositional Zero-Shot Learning

Effectiveness of SAD-SP: Among methods predicting simple primitives in Table I, our proposed SAD-SP achieves the best performance on all criteria for OW-CZSL. SAD-SP outperforms SOTA methods in HM and AUC. Compared with the best SOTA method (i.e., KG-SP), SAD-SP relatively improves HM by 5.4% (MIT-States 7.4 vs 7.8), 4.0% (UT-Zappos 42.3 vs 44.0), 25.5% (C-GQA 4.7 vs 5.9) and increases AUC by 7.7% (MIT-States 1.3 vs 1.4), 7.2% (UT-Zappos 26.5 vs 28.4), 28.5% (C-GQA 0.78 vs 1.002), respectively. Note that KG-SP applies external knowledge to train a concept network to eliminate some impossible compositions. The consistent improvement of HM and AUC indicates that knowledge disentanglement and semantic attention in SAD-SP can effectively provide meaningful semantic information related to feasibility and contextuality in compositional predictions, which can be as effective as external semantic knowledge. SAD-SP achieves the best S and U scores except SAD-SP obtaining the highest S on C-GQA. In other words, when applying extremely large or small bias terms, constraining the output range into seen or
unseen compositions, our method can have the best ability to fit seen compositions and to be generalized to unseen compositions.

SAD-SP consistently achieves the highest scores on UT-Zappos and C-GQA compared to methods that project the input into a shared embedding space. When compared with the best state-of-the-art methods (i.e., Co-CGE), SAD-SP exhibits relative improvements in HM by 7.8% (UT-Zappos 40.8 vs 44.0) and 22.9% (C-GQA 4.8 vs 5.9), and in AUC by 21.9% (UT-Zappos 23.3 vs 28.4) and 28.5% (C-GQA 0.78 vs 1.002). For both datasets, models based on compositional embeddings generally perform worse than models predicting simple primitives, with the exception of Co-CGE. On MIT-States, CompCos and Co-CGE outperform SAD-SP. However, improvements in CompCos are dataset-specific. Co-CGE continues to show competitive performance on C-GQA while CompCos significantly decreases its performance. This suggests that using feasibility, computed by pairwise cosine similarity of label embeddings, can effectively reduce the inherent label noise on MIT-States [20]. However, merely relying on cosine feasibility may not be effective for other datasets. Graph structure and cosine feasibility can provide complementary effects across diverse datasets. Considering that Co-CGE depends on varying external knowledge sources for different datasets [22] to aid in learning embeddings, such as word2vec [64] for C-GQA and word2vec+fasttext [65] for MIT-States, SAD-SP exhibits competitive performance across datasets without requiring additional semantic knowledge, demonstrating the effectiveness of our balanced hyperparameters and merging way.

Effectiveness of SA and KD: Compared with baselines (i.e., LE+ and VisProd), balancing seen and unseen compositions can boost the performance, e.g., VisProd++. Merely increasing network depth does not ensure improvements in model performance. For example, Ensemble surpasses VisProd++ on MIT-States and C-GQA but underperforms on UT-Zappos. Learning contextuality or feasibility can enhance model performance, e.g., SAD-SP surpasses Ensemble on three datasets. Moreover, the models using external semantic knowledge (i.e., CompCos, CGE, Co-CGE, and KG-SP) tend to have better performance than other methods following the same manner of recognizing compositions. The external knowledge is usually used as semantic experts who project words of labels into graph embedding or help eliminate infeasible compositions. The effectiveness of external knowledge suggests that a significant factor in recognizing compositions precisely is to capture the semantic relations between primitive concepts and their combinations. Different from these models, SAD-SP does not rely on external knowledge except the commonly used backbone trained on ImageNet, but SA and KD can still improve the model ability as effectively as the external knowledge. In other words, SA and KD can learn more semantic information that remains unused yet informative than conventional methods. We also exhibit the performance without end-to-end training, i.e., CGEṭ, VisProdṭ++, KG-SPṭ, and SAD-SPṭ. We can observe that SAD-SPṭ defeats other methods in HM and AUC on three datasets. It indicates that SA and KD have the strongest discriminative ability when they are applied on top of the same backbones.

### Table II

| Combinations | SAD-SPṭHM | SAD-SPṭAUC | MeanHM | MeanAUC |
|--------------|-----------|------------|--------|---------|
| Own(SP+SA+KD) | 8.334 | 1.315 | 8.969 | 1.762 | 8.652 | 1.639 |
| SP+SA+KD | 8.096 | 1.367 | 8.677 | 1.558 | 8.387 | 1.463 |
| SP+SA | 8.215 | 1.446 | 8.571 | 1.599 | 8.393 | 1.523 |
| SP+SA+KD | 7.837 | 1.409 | 8.922 | 1.656 | 8.360 | 1.533 |
| (SP+SA)+KD | 8.317 | 1.464 | 8.233 | 1.667 | 8.225 | 1.474 |
| (SP+KD)+SA | 8.311 | 1.499 | 8.315 | 1.541 | 8.413 | 1.520 |
| (SP+KD)+SA | 8.291 | 1.498 | 8.125 | 1.514 | 8.208 | 1.506 |

Effectiveness of SA Network Architecture: As shown in Table I, the GT version of SAD-SP underperforms the Prediction version in AUC, except for the fixed-backbone setting on C-GQA. Given that SP predictions can convey more class information dependent on instances to SA than GT labels, using feature representations with higher instance-specific information as inputs may benefit SA training. The Prediction-variant exhibits competitive performance compared to our method when using end-to-end training, but it underperforms on UT-Zappos and C-GQA datasets by 0.7 and 0.23 in AUC when using a fixed backbone. This indicates that the second-last layer output and the last-layer output may convey similar information to SA when the network is fine-tuned on datasets. However, when the network is not fully fine-tuned on datasets, the uncompressed embedding may convey more meaningful information to SA than the highly compressed embedding. This result validates the effectiveness of our proposal, i.e., employing the second-last layer output to represent the predicted class information. Our method enables SA to learn more semantic information from inputs, thereby enhancing model performance.

C. Methods and Network Architecture Validation

1) Validation of Merging Ways: We regard multiplication and addition as two straightforward yet effective methods for merging modules. Consequently, we validate our merging strategy, namely the weighted summation, by evaluating all eight combinations of modules employing multiplication and addition. As weight hyperparameters in multiplication do not impact the predictions of module products, we exclusively utilize hyperparameters in addition. We maintain similar contributions from each module in the final predictions to examine the effects of varying merging approaches.

Specifically, for combinations involving addition, we retain the same hyperparameters for modules as in our weighted summation. The key difference is that we employ the summed weights for the product of two modules and divide the weight by half if a module is involved in two module products. For instance, we use γ1+γ3 as the weight for SA+KD in SP+(SA+KD), and (γ1/2+γ2/2, γ1/2+γ2/2) for (SA,KD) in SP+(SA+KD) respectively.

As shown in Table II, our method only using addition to merge modules consistently achieves the highest scores on both HM and AUC metrics on the validation set of the MIT-States dataset. In contrast, only using multiplication without parameters show
relatively lower performance. The strategy of merging SP with the combination of (SA, KD) achieves relatively higher performance than incrementally merging SP with other modules. The results demonstrate the importance of effectively integrating all three modules and highlight the superiority of our proposed method over alternative merging combinations.

2) Validation of Regularization Terms: We conduct an ablation study to evaluate the effectiveness of our proposed two regularization terms for KD and SA respectively, namely $\mathcal{L}_{dis}$ and $\otimes$ in $\mathcal{L}_{att}$. We compare our models with and without these two regularization terms under the same parameter settings, as shown in Table III. The results demonstrate that both terms contribute to improved performance in terms of HM and AUC. When both terms are disabled, the model yields the worst performance. Disabling $\mathcal{L}_{dis}$, has a more significant impact on the fine-tuned version of SAD-SP, while disabling $\otimes$ in $\mathcal{L}_{att}$ has a greater effect on the fixed backbone version. These results suggest that both regularization terms enhance the model’s generalization ability when handling generalized CZSL tasks.

3) Validation of SA and KD: In this section, we present an ablation study of removing modules, namely SA and KD, on SP predictions; branch-level studies further investigate the impact of varying the number of layers replacing the original two-layer FCN in $f_3$, on the MIT-States validation set, validating the balance of our chosen hyperparameters, i.e., $\gamma_1 = 0.7$, $\gamma_2 = 0.25$, and $\gamma_3 = 0.05$. We first optimize SAD-SP with or without end-to-end training and obtain the optimized model. Then, we vary $\gamma_2$ and $\gamma_3$ within the range of $[0.05, 0.5]$ with a step of 0.05 to measure the AUC scores for different hyperparameter settings, as shown in Fig. 3. We observe that most hyperparameter settings that keep SP as main components, including ours, exhibit similar high performances around 1.53 and 1.7 for the fixed backbone and the fine-tuned backbone, respectively. This validates that our hyperparameters are balanced on the MIT-States validation set. Additional experiments exhibiting the balance of our hyperparameters on the testing sets can be found in the Appendix, available online.

5) Validation of Extractor Structure: In Table V, we investigate the impact of varying the number of layers replacing the original two-layer FCN in $f_3$ on the MIT-States validation set. Beginning with a single-layer extractor, we incrementally add a layer beneath it each time, keeping the output dimension as 1024. The AUC and HM scores demonstrate the best performance when employing two layers. Versions with a fixed backbone underperform compared to those with a fine-tuned backbone, suggesting that a fixed backbone may not be able to effectively extract visual information. This observation aligns with findings from other related studies [21], [23], indicating that a fine-tuned backbone and a 2-layer extractor show the best performance when handling compositional zero-shot learning tasks across these three datasets.

D. Feasibility and Contextuality Analysis

In this section, we conduct a comprehensive analysis of the feasibility and contextuality learned by our model through experiments on the test sets. We examine the contributions of each component by performing a component analysis, assess the learned feasibility by visualizing the overall distribution and the most/least prominent compositions of semantic attention, evaluate the effectiveness of KD in learning contextuality through embedding visualization, and illustrate how our models refine predictions using a case study.

1) Component Analysis: In this section, we conduct component analysis from a module-level and branch-level perspective, analyzing the effects of feasibility and contextuality on performance. Module-level studies reveal the effects of modules, i.e., SA and KD, on SP predictions; branch-level studies further analyze the detailed effects of SA and KD on model performance.

### Table III

| Loss Variants | SAD-SP$_{H}$ | SAD-SP$_{L}$ |
|---------------|---------------|---------------|
| Ours          | 8.334         | 1.515         |
| w/o $\mathcal{L}_{dis}$ | 8.097         | 1.435         |
| w/o $\otimes$ in $\mathcal{L}_{att}$ | 7.945         | 1.385         |

Table III: Ablation Study on Loss Variants Without Regularization Terms in KD and SA on the Validation Set of the MIT-States Dataset

### Table IV

| Model      | Combination | SF$^a$ | Combination$_b$ | SF$^b$ |
|------------|-------------|--------|-----------------|--------|
| SAD-SP$_{H}$ |             |        |                 |        |
| w/o SA     | 1.633       | 8.546  | 1.635           | 8.994  |
| w/o KD     | 1.632       | 8.623  | 1.619           | 8.643  |
| Ensemble   | 1.554       | 8.547  | 1.123           | 8.236  |

Table IV: Ablation Study of Modules on the Validation Set of the MIT-States Dataset

### Table V

| Output Dimension | Layer 1 | Layer 2 | Layer 3 | Layer 4 |
|------------------|---------|---------|---------|---------|
|                  | +1024   | -756   | +512   | +256   |

Table V: AUC and HM Scores of Different Layer Structures With Fixed or Fine-Tuned Backbones on the Validation Set of MIT-States

*The highest scores are highlighted in bold.*
Module-Level Analysis: In Table VI, we disable SA and KD in SAD-SP, and their corresponding predictors in Ensemble to exhibit the performance of module-level variants. SP in SAD-SP represents the results of the enhanced VisProd++ branch by our SA and KD modules; SA-SP and KD-SP are variants disabling the modules of KD and SA, respectively. When comparing 1-layer Ensemble with VisProd++, Ensemble shows worse underperforms on three datasets, indicating that directly assembling multiple predictors could diminish the SP performance. Conversely, SP in SAD-SP shows a competitive or superior performance compared to VisProd++, suggesting that our methods have complementary learning effects on the backbone and shared state/object extractors, which preserves and improves the discrimination ability of SP. Comparing SAD-SP variants and their corresponding Ensemble variants, we can observe that Ensemble variants of 1&2 Layer and 1&3 Layer with end-to-end training show better performance than SA-SP and KD-SP. However, SAD-SP still increases the best performance of Ensemble variants by 0.033/0.197 in AUC/HM. SAD-SP also exhibits significantly better performance in other testing settings, where the lowest-performing SAD-SP variant surpasses the best Ensemble variant. These results indicate that directly employing multiple predictors with a deeper network structure does not guarantee improvements in model performance and could be detrimental to the original 1-layer predictor. The deeper network structure is not the primary contributor to the overall improvement of our methods. Our contextuality and feasibility learning enhancing the model performance over Ensemble in most cases validate the model effectiveness and their complementary effects to the SP module.
Moreover, we can observe that SAD-SP achieves the best AUC scores across variants of SAD-SP on MIT-States, MIT-States\textsubscript{St} and UT-Zappos\textsubscript{St}. It also achieves the highest HM scores in the corresponding datasets, indicating that both SA and KD can benefit the compositional recognition in a complementary way on MIT-States and UT-Zappos with or without end-to-end training. Compared with SP, KD-SP consistently improves model performance on different datasets, especially by a large margin on UT-Zappos and C-GQA (e.g., HM score: 43.441 vs 44.068 and 5.795 vs 6.086). The consistent improvement of KD indicates the effectiveness and universality of knowledge disentanglement on learning contextuality in OW-CZSL. SA-SP can increase the AUC and HM scores on C-GQA but may impair model performance on MIT-States and UT-Zappos. The demerit may be caused by the label noise on MIT-States and weak semantic feasibility relations between shoes and materials on UT-Zappos. Nonetheless, the feasibility information learned by SA can boost KD under most conditions.

**Branch-Level Analysis With End-to-End Training:** We exhibit AUC and HM scores by disabling branches within SAD-SP in Table VII. \( p_f(s|o) \) and \( p_c(s) \) denote probabilistic revisions of feasibility and contextuality provided by semantic attention and knowledge disentanglement for the state branch. Similarly, \( p_f(o|s) \) and \( p_c(o) \) are the corresponding revisions for the object branch. On MIT-States and MIT-States\textsubscript{St}, we can observe that disabling any branch in SAD-SP will lead to a lower HM or AUC score. This indicates that both semantic attention and knowledge disentanglement in the state or object branch can provide complementary information for compositional recognition. Comparing feasibility and contextuality branches, we can observe that the learned feasibility (i.e., \( p_f(o|s) \) and \( p_f(s|o) \)) are more informative than the learned contextuality (i.e., \( p_c(o) \) and \( p_c(s) \)). Disabling \( p_f(o|s) \) or \( p_f(s|o) \) may cause severe information loss while removing contextuality causes the least information loss. For example, on MIT-States, disabling \( p_f(o|s) \) results in decreasing AUC up to 0.090 while disabling \( p_f(s|o) \) & \( p_c(s) \) achieves the highest AUC increase, i.e., 0.004, compared with disabling \( p_f(s|o) \). On MIT-States\textsubscript{St}, disabling \( p_f(o|s) \) results in decreasing HM up to 0.141 while disabling \( p_f(o|s) \) & \( p_c(s) \) achieves the highest HM increase, i.e., 0.048, compared with disabling \( p_f(o|s) \).

Conversely, knowledge disentanglement plays a more important role than semantic attention on UT-Zappos. Eliminating contextuality will cause a significant performance decrease, especially state contextuality. AUC drops by up to 2.298/1.329 and HM declines by up to 0.819/0.612 with/without end-to-end training due to disabling \( p_c(s) \), indicating that learning unbiased material representations is a key factor in recognizing shoe compositions precisely. Feasibility seems to be less effective on UT-Zappos because disabling state or object attention has a limited effect on the model performance. However, SA can still provide some complementary information to boost KD performance when not using end-to-end training.

Though SAD-SP outperforms SOTA methods on C-GQA, disabling \( p_c(s) \) and \( p_f(o|s) \) can further enhance SAD-SP to achieve better performance. For example, disabling knowledge disentanglement of state in the end-to-end training can relatively increase AUC by 4.3% (1.002 vs 1.045) and HM by 3.7% (5.877 vs 6.094); removing semantic attention of object in the non-end-to-end training relatively raises AUC by 2.5% (0.856 vs 0.877) and HM by 1.9% (5.140 vs 5.237). The improvements indicate that SAD-SP sometimes fails to learn the most balanced feasibility- and contextuality-dependence on C-GQA; however, SAD-SP still has a strong ability to capture the semantic information of dependence between simple primitives.

2) **Feasibility Distribution of Semantic Attention:** In this section, we analyze the feasibility (i.e., weight) distributions of SA, showing the learned semantic relations in compositional feasibility from a comparative perspective. We first accumulate the instance-specific object and state attention according to their ground-truth labels, obtaining the dataset-level attention map. Then, we use min-max normalization to normalize the attention map conditioned on the state or object to learn the comparative relations.

Formally, given an arbitrary input, let the \( i^{th} \) object \( o_i \) and the \( j^{th} \) state \( s_j \) be the ground-truth labels of the input, we accumulate attention to learn a matrix \( \mathcal{M} \) representing the weight distribution as follows:

\[
\mathcal{M}_i = Softmax(\mathcal{M}_i + a^o) \\
\mathcal{M}^j_i = Softmax(\mathcal{M}^j_i + a^s)
\]

where \( \mathcal{M}_i \in [0, 1]^{\text{S} \times \text{O}} \) is the weight matrix of attention initialized as a zero matrix; \( a^o \) and \( a^s \) represent the learned attention vector for the object and the state; the \( i^{th} \) row \( \mathcal{M}_i \) and the \( j^{th} \) column \( \mathcal{M}^j_i \) in the weight matrix represent the accumulated attention weights for the \( i^{th} \) object \( o_i \) and the \( j^{th} \) state \( s_j \), respectively.

After obtaining the dataset-level weight matrix, we take a few groups of similar objects and their related states from MIT-States as an example to illustrate the effectiveness of SA. We normalize the selected weights along with the axis of the object or state in

| Disable | MIT-States | MIT-States\textsubscript{St} | UT Zappos | UT Zappos\textsubscript{St} | C-GQA | C-GQA\textsubscript{St} |
|---------|------------|-----------------|-----------|-----------------|-------|------------------|
| AUC     | HM         | AUC             | HM        | AUC             | HM    | AUC              | HM    |
| \( p_f(s|o) \) | 1.281 | 7.766 | 1.153 | 7.055 | 28.117 | 43.874 | 22.857 | 39.452 | 0.992 | 5.837 | 0.837 | 5.163 |
| \( p_f(o|s) \) | 1.272 | 7.746 | 1.164 | 7.033 | 28.265 | 43.825 | 23.808 | 39.661 | 0.994 | 5.921 | 0.877 | 5.237 |
| \( p_c(o) \) | 1.289 | 7.793 | 1.162 | 7.111 | 28.267 | 42.767 | 22.299 | 39.418 | 1.045 | 6.094 | 0.846 | 5.138 |
| \( p_f(s|o) \)\&\( p_c(s) \) | 1.283 | 7.836 | 1.166 | 7.059 | 27.831 | 43.483 | 22.774 | 39.819 | 0.994 | 5.982 | 0.869 | 5.136 |
| \( p_f(o|s) \)\&\( p_c(o) \) | 1.274 | 7.786 | 1.152 | 7.044 | 25.819 | 42.545 | 22.051 | 38.876 | 1.008 | 5.863 | 0.814 | 5.046 |
| \( p_f(s|o) \)\&\( p_c(s) \) | 1.285 | 7.827 | 1.144 | 6.998 | 27.755 | 43.266 | 22.465 | 39.429 | 0.967 | 5.633 | 0.874 | 5.129 |
| \( p_f(o|s) \)\&\( p_c(o) \) | 1.270 | 7.719 | 1.152 | 7.081 | 28.070 | 42.363 | 22.181 | 39.173 | 1.013 | 5.915 | 0.894 | 5.119 |
| \( p_f(s|o) \)\&\( p_c(s) \) | 1.284 | 7.759 | 1.165 | 7.092 | 27.799 | 43.647 | 22.679 | 39.424 | 1.000 | 5.912 | 0.867 | 5.117 |

The best results are in bold.
Fig. 4, showing the learned semantic relations between objects and states. Each row in Fig. 4(a) can be viewed as the state-conditioned object attention; each column in Fig. 4(b) can be viewed as the object-conditioned state attention.

We can observe that SA tends to assign heavy weights to simple primitives that share similar appearances. For example, in the object attention, peeled has a high weight score, i.e., feasibility, to fruit, apple, and banana; dry relates to sea and river closely. In the state attention, vegetables are highly feasible to show a state of diced, mashed, peeled, sliced, or ripe; animals and cats show close correlations to huge, large, and old. Obviously, the high feasibility tends to propagate within the same types of objects or states, proving that SA is effective in learning similarity-driven semantics in the datasets. Moreover, no objects have high feasibility to multiple states in the object attention, but large and old obtain the high feasibility across multiple objects in the state attention. It makes sense because objects may not be described by many types of states while some states (e.g., large) can describe most objects in the real world. Large and old tend to show high feasibility to the same objects, suggesting that many large objects can be also described by old. It is common, especially when describing animals, e.g., larger cats usually equal older cats.

3) Feasible Compositions in Semantic Attention: In this section, we demonstrate the effectiveness of SA by showing the most/least feasible compositions based on the frequency. We argue that the more frequently a composition is assigned the highest/lowest weight in the attention map, the more/less feasible it is considered by the attention mechanism. Therefore, we count the frequency and obtain the most frequent state-object pairs, representing the most/least feasible compositions, i.e., the most frequent compositions with the highest/least weight in $p_f(s|o)$ and $p_f(o|s)$, for each state and object. We take MIT-States and C-GQA datasets as examples, showing the Top/Bottom-3 feasible compositions (i.e., Top/Bottom-3 most feasible compositions with the highest/lowest attention weight).

In Table VIII, we exhibit the Top-3 feasible compositions conditioned on objects and states. We show the most feasible compositions in two different settings: the Open-World space (OW) and the Unseen compositional (UC) space. OW may contain compositions from seen compositions while UC only contains unseen compositions. Thus, we can know whether SA can infer feasible unseen compositions by only learning the existing semantics of datasets. We can observe that SA effectively finds the GT unseen compositions and views them as the one of most feasible compositions. For example, the GT unseen composition, i.e., straight blade, is viewed as a Top-3 feasible composition for straight under both OW and UC settings. Blue tail, hairy tail, and long tail are the most feasible compositions for a tail while only long tail is seen during training. These findings prove that SA can predict GT unseen compositions by learning the seen semantics in datasets.

In Table IX, we can observe that the most infeasible compositions have a low probability of existing in the real world, e.g., bent flame and feathered cauliflower. There are two possible reasons that SA considers these compositions infeasible. 1) The related images of simple primitives are too distinct from other images. For example, seen images that relate to flame are molten flame and brushed flame. These distinct images share little similarity with other images. 2) Few related seen compositions exist in datasets. For example, feathered only relates to wing during training. SA propagates little feasibility based on feathered and thus feathered has no close correlations to other simple primitives. The infeasible compositions suggest that our method is capable of finding those distinct simple primitives, and then assigning low feasibility to their unrelated concepts.

4) Unbiased Features of Knowledge Disentanglement: In this section, we project features to two-dimensional embeddings based on t-SNE [71]. Then, we visualize the projected embeddings of original features and disentangled features provided by KD, showing whether the distributional loss $L_{kd}$ can supervise KD to disentangle features. In Figs. 5 and 6, we view the mean
TABLE VIII
EXAMPLES OF TOP-3 FEASIBLE COMPOSITIONS IN SA ON MIT-STATES AND C-GQA DATASETS, WHERE OW AND UC REPRESENT THE LEARNED FEASIBLE COMPOSITIONS EXISTING IN THE OPEN-WORLD (OW) COMPOSITIONAL SPACE OR UNSEEN COMPOSITIONAL (UC) SPACE; GT DENOTES THE GROUND-TRUTH (GT) UNSEEN COMPOSITIONS IN THE TESTING SET; COMPOSITIONS EXISTING IN THE TRAINING SET ARE IN ITALICS

| State   | MIT-States | Object | Top-3 Feasible States |
|---------|------------|--------|-----------------------|
| **straight** | OW: road, sword, blade | velvet | OW: brushed, crushed, wrinkled |
|         | UC: blade, bronze, highway |         | UC: wrinkled, creased, crumpled |
|         | GT: blade, highway, pool |         | GT: crumpled, wrinkled |
| **squished** | OW: sandwich, tomato, bread | blade | OW: blunt, large, straight |
|         | UC: bread, fish, plate |         | UC: large, straight, bent |
|         | GT: bread, bus, coin, fish, penny |         | GT: bent, narrow, straight |

C-GQA

| State   | MIT-States | Object | Top-3 Feasible States |
|---------|------------|--------|-----------------------|
| **forested** | OW: hill, mountain, tree | mattress | OW: crumpled, folded, blue |
|         | UC: hill, cliff, forest |         | UC: crumpled, folded, carpeted |
|         | GT: hillside, hill |         | GT: red, soft, folded |
| **asian** | OW: boy, building, person | tail | OW: blue, hairy, long |
|         | UC: building, person, bleachers |         | UC: blue, silver, orange |
|         | GT: person |         | GT: blue, silver, orange |

The overlapped simple primitives are in bold.

TABLE IX
EXAMPLES OF BOTTOM-3 FEASIBLE COMPOSITIONS ON MIT-STATES AND C-GQA DATASETS, AND THE CORRESPONDING TOP-3 FEASIBLE SEEN COMPOSITION EXAMPLES RELATING TO THE FOUND INFEASIBLE OBJECTS/STATES

| State | MIT-States | Object | Top-3 Seen States |
|-------|------------|--------|-------------------|
| **lightweight** | flame, vacuum, drum, flame, vacuum, flame, laptop | flame laptop | OW: black, open, silver |
| **Object** | Bottom-3 Feasible States | Infeasible State | Top-3 Seen Objects |
| book | standing, dull, short | standing | tower |
| bucket | blunt, mashed, standing | mashed | bean, vegetable, banana |

C-GQA

| State | MIT-States | Object | Top-3 Seen States |
|-------|------------|--------|-------------------|
| **artificial** | shield, charger, courtyard | shield case | glass, protective |
| **Object** | Bottom-3 Feasible States | Infeasible State | Top-3 Seen Objects |
| cauliflower | feathered, rustic, winding | feathered | wing |
| eagle | connected, discolored, miniature | connected | chain, cord |

Fig. 5. Original feature embedding and disentangled feature embedding for synthetic ankle shoes and leather ankle shoes. Figures (a–b) plot the original and disentangled object embeddings for ankle shoes, respectively. Figures (c–d) plot the original and disentangled state embeddings for synthetic and leather, respectively.

embedding of simple primitives as a prototype to show the center point of embedding distributions, and we exhibit two pairs of compositions on UT-Zappos. Fig. 5(a)–(d) visualize the composition pair synthetic ankle shoes and leather ankle shoes, which share the same object (i.e., ankle shoes) but different states (i.e., synthetic and leather). In Fig. 5(a)–(b), KD can reduce the distance of the object prototype of ankle shoes significantly. Thus, the embedding space of ankle shoes is more concentrated after disentanglement. Fig. 5(c)–(d) share a similar prototype distance, but nodes are more dispersed after disentanglement. In Fig. 6(a)–(d), we plot compositions that share the same state but different objects, i.e., synthetic ankle shoes and synthetic mid-calf boots. From Fig. 6(a)–(b), KD can increase the distance of object prototype of ankle shoes and mid-calf boots. The node distributions of ankle shoes and mid-calf boots are more concentrated around their prototype after disentanglement. Fig. 6(c)–(d) show that the distance between the state prototypes of synthetic is largely reduced, which means that KD can effectively extract the state information from different objects. In conclusion, KD supervised by $L_{kd}$ can effectively disentangle states/objects to
learn unbiased feature representations. The unbiased feature representations of the same state/object in different compositions are clustered; the feature distributions of different objects/states are dispersed. Thus, KD can learn more disentangled and unbiased features for classification than the original features, which may ease the biased predictions caused by contextuality.

5) Case Study: We conduct a case study to illustrate how SAD-SP revises the predictions of SP and the limitations of our model in Fig. 7. In the positive cases, SA can calibrate steel to screw and sitting to white, which is more consistent with the semantic relations in the dataset. KD enables SAD-SP to identify the moss on the tree and to refine animals to giraffes, which indicates the improvements in the ability to recognize simple primitives in our model. Moreover, the synergy of SA and KD can fix some misclassifications in SP, SA-SP, and KD-SP, such as modifying a cracked computer into a new laptop and letting the model focus on the wooden material of a bed. However, when there are multiple simple primitives existing in the image, our model tends to make wrong predictions. For example, our model fails to refine an animal as a cat when the cat is hidden in the grass, and an insect flies around it. It also views the blue color from the background as the state of the main object mistakenly. These mistakes indicate that our model sometimes fails to locate the main object or state in the image. The possible reason is that the model and annotator view different things as the main object in the image. While SAD-SP can partially ease this issue, e.g., SA improves red into indoor, or SAD-SP enhances red into wood, it still classifies a knit hat as a purple hat, a person with long hair as a lady, and a thick book as a closed book. These predictions may not be mistakes strictly, but it shows the deficiency of our model in learning the human preference for label annotations, e.g., SAD-SP is not aware that color may have the lowest priority when describing objects.

6) Summary and Discussion: Summary of experiments: Compared with the SOTA methods in OW-CZSL, SAD-SP shows competitive or better performance using fine-tuned and fixed backbone networks. We propose two new parallel networks, SA and KD, which can enhance SP by learning feasibility- and contextuality-dependence. We demonstrate the effectiveness of SA and KD from multiple aspects: 1) We perform extensive experiments on the validation set, e.g., validating our regularization terms, modules, and hyperparameters, to demonstrate the effectiveness and balanced robustness of our methods; 2) We analyze the contributions of each module and branch to the final predictions on the test sets, showcasing the enhancements brought by feasibility and contextuality during inference; 3) We visualize the learned attention weights and disentangled feature representations, suggesting that our SA can effectively infer unseen compositions based on existing semantic relations, while KD can learn unbiased disentangled features to achieve a better feature distribution; 4) We conduct a case study to illustrate that SA and KD can effectively refine the incorrect predictions of SP and partially learn the priority present in manual annotations.

Discussion of Datasets: From the experiments, we can observe that the advances of SAD-SP on datasets are owing to different modules. Since we use the same setting for training, e.g., network structures, learning rates, and hyperparameters, we consider the learning abilities of our models to be similar. We argue that challenges constraining the progress of SOTA methods...
are different in datasets. UT-Zappos is a distinct dataset that only contains shoes and common materials. Since no strict limitations of materials are posed on making shoes, feasibility-dependence provides little information for improving the predictions. On the contrary, contextuality-dependence is essential to learn, which plays a vital role in discriminating entangled shoe types and materials. This is why KD contributes the most to SAD-SP on UT-Zappos in identifying footwear components. Different from UT-Zappos, MIT-States and C-GQA datasets consist of objects from a wide range, e.g., animals and buildings. Due to the diverse objects, feasibility-dependence is informative and essential to the performance improvement of SAD-SP. Meanwhile, the wider range of objects and related states makes it more difficult to recognize simple primitives, especially states. On MIT-States, our model is still capable of learning the unbiased states and objects beneficial for predictions. However, on C-GQA, the current network structure may not handle the extremely large scope of simple primitives (~4 times the states and ~2 times the objects of MIT-States) well, leading to SAD-SP only achieving the best performance after disabling some branches. We speculate that datasets with more diverse simple primitives require stronger learning capabilities of feasibility- and contextuality-dependence.

Limitations: Though branches in SA and KD can achieve robust results under a unified setting, the fixed hyperparameters result in SAD-SP being unable to balance feasibility and context dependencies dynamically. For example, SAD-SP cannot disable invalid branches dynamically may result in a sub-optimal output. In addition, while SA and KD can alleviate some biased predictions in SP, such as refining a vague concept into an explicit class and denoising the irrelevant visual information in the background, SAD-SP may still misunderstand the appropriate simple primitives or the main object in the images.

V. CONCLUSION

In this work, we propose a Semantic Attention and knowledge Disentanglement guided Simple Primitives (SAD-SP) to tackle the deficiency of feasibility and contextuality in simple primitives under the open-world compositional zero-shot learning setting. We design semantic attention, which learns to infer the feasibility relations in datasets driven by visual similarity, to provide auxiliary classification information for predictions. We also propose a generative knowledge disentanglement module, which conducts knowledge disentanglement supervised by a distributional loss, to learn unbiased feature representations specific to states and objects, easing the biased predictions caused by contextuality. Via experiments, we discuss the underlying reasons for our improvement on different datasets, providing some insightful analysis for the community. We conclude some potential challenges and limitations in OW-CZSL, e.g., the limited semantic relations between shoes and materials on UT-Zappos, the difficult state recognition on MIT-States and C-GQA, and the difficulty in learning the annotation priority of humans. In the future, we plan to implement dynamic γ selections and enhance the network structure of the backbone [72], [73], which may fix the learning deficiency and balance the learning tendencies dynamically. Given the superior performance on MIT-States and the competitive performance on C-GQA of Co-CGE, it is promising to incorporate graph structure information and cosine similarity between embeddings within simple primitive methods. However, the approach could enhance the discriminative capabilities when recognizing datasets with various states and objects.

REFERENCES

[1] Y. Li, Z. Liu, L. Yao, and X. Chang, “Attribute-modulated generative meta learning for zero-shot learning,” IEEE Trans. Multimedia, vol. 25, pp. 1600–1610, 2021.
[2] H. Zhao, S. Guo, and Y. Lin, “Hierarchical classification of data with long-tailed distributions via global and local granulation,” Inf. Sci., vol. 581, pp. 536–552, 2021.
[3] Z. Liu, Y. Li, L. Yao, X. Wang, and G. Long, “Task aligned generative meta-learning for zero-shot learning,” in Proc. 55th AAAI Conf. Artif. Intell., 2021, pp. 8723–8731.
[4] Y. Zhang, X.-S. Wei, B. Zhou, and J. Wu, “Bag of tricks for long-tailed visual recognition with deep convolutional neural networks,” in Proc. AAAI Conf. Artif. Intell., 2021, pp. 3447–3455.
[5] D. Samuel, Y. Atzmon, and G. Chechik, “From generalized zero-shot learning to long-tail with class descriptors,” in Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis., 2021, pp. 286–295.
[6] C. Yan, X. Chang, M. Luo, H. Liu, X. Zhang, and Q. Zheng, “Semantics-guided contrastive network for zero-shot object detection,” IEEE Trans. Pattern Anal. Mach. Intell., early access, Jan. 4, 2022, doi: 10.1109/TPAMI.2021.31614875.
[7] F. Pourpanah et al., “A review of generalized zero-shot learning methods,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 45, no. 4, pp. 4051–4070, Apr. 2023.
[8] B. Romera-Paredes and P. Torr, “An embarrassingly simple approach to zero-shot learning,” in Proc. Int. Conf. Mach. Learn., 2015, pp. 2152–2161.
[9] W. Wang, V. W. Zheng, H. Yu, and C. Miao, “A survey of zero-shot learning: Settings, methods, and applications,” ACM Trans. Intell. Syst. Technol., vol. 10, no. 2, p. 1–37, 2019.
[10] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, “Generalizing from a few examples: A survey on few-shot learning,” ACM Comput. Surv., vol. 53, no. 3, pp. 1–34, 2020.
[11] T. Hospedales, A. Antoniou, P. Micaelli, and A. Torokney, “Meta-learning in neural networks: A survey,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 44, no. 9, pp. 5149–5169, Sep. 2021.
[12] F. Ruis, G. Burghouts, and D. Bucur, “Independent prototype propagation for zero-shot compositionality,” in Proc. Adv. Neural Inf. Process. Syst., 2021, pp. 10 641–10 653.
[13] 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.
[14] A. Ramesh et al., “Zero-shot text-to-image generation,” in Proc. Int. Conf. Mach. Learn., 2021, pp. 8821–8831.
[15] S. Changpin, W.-L. Chao, B. Gong, and F. Sha, “Synthesized classifiers for zero-shot learning,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2016, pp. 5327–5336.
[16] T. Nagarajan and K. Grauman, “Attributes as operators: Factorizing unseen attribute-object compositions,” in Proc. Eur. Conf. Comput. Vis., 2018, pp. 169–185.
[17] S. Purushwalkam, M. Nickel, A. Gupta, and M. Ranzato, “Task-driven modular networks for zero-shot compositional learning,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2019, pp. 3593–3602.
[18] I. Misra, A. Gupta, and M. Hebert, “From red wine to red tomato: Composition with context,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 1792–1801.
[19] Y.-L. Li, Y. Xu, X. Mao, and C. Lu, “Symmetry and group in attribute-object compositions,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 3593–3602.
[20] Y. Atzmon, F. Kreuk, U. Shalit, and G. Chechik, “A causal view of compositional zero-shot recognition,” in Proc. Adv. Neural Inf. Process. Syst., 2020, pp. 1462–1473.
[21] S. Karthik, M. Mancini, and Z. Akata, “KG-SP: Knowledge guided simple primitives for open world compositional zero-shot learning,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2022, pp. 9336–9345.
[22] M. Mancini, M. F. Naeem, X. Yuan, and Z. Akata, “Learning graph embeddings for open world compositional zero-shot learning,” IEEE Trans. Pattern Anal. Mach. Intell., early access, Mar. 30, 2022, doi: 10.1109/TPAMI.2022.3163667.

[23] S. Karthik, M. Mancini, and Z. Akata, “Revisiting visual product for compositional zero-shot learning,” in Proc. Workshop Distribution Shifts: Connecting Methods Appl., 2021.

[24] D. Huynh and E. Elhamifar, “Compositional zero-shot learning via fine-grained dense feature composition,” in Proc. Adv. Neural Inf. Process. Syst., 2020, pp. 19849–19860.

[25] M. Yang, C. Xu, A. McCallum, and C. Deng, “A decomposable causal view of compositional zero-shot learning,” IEEE Trans. Multimedia, early access, Aug. 22, 2022, doi: 10.1109/TMM.2022.3200578.

[26] X. Li, X. Yang, K. Wei, C. Deng, and M. Yang, “Siamese contrastive embedding network for compositional zero-shot learning,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognition, 2022, pp. 9326–9335.

[27] T. Zhang, K. Liang, R. Du, X. Sun, Z. Ma, and J. Guo, “Learning invariant visual representations for compositional zero-shot learning,” 2022, arXiv:2206.00415.

[28] N. Saini, K. Pham, and A. Shrivastava, “Disentangling visual embeddings for attributes and objects,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognition, 2022, pp. 13 658–13 667.

[29] M. Mancini, M. F. Naeem, Y. Xian, and Z. Akata, “Open world compositional zero-shot learning,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognition, 2021, pp. 5222–5230.

[30] M. F. Naeem, E. P. Ornek, X. Yian, L. Van Gool, and F. Tombari, “3D compositional zero-shot learning with decompositional consensus,” in Proc. Eur. Conf. Comput. Vis., 2022, pp. 713–730.

[31] Z. Hou, X. Peng, Y. Qiao, and D. Tao, “Visual compositional learning for human-object interaction detection,” in Proc. Eur. Conf. Comput. Vis., 2020, pp. 584–600.

[32] Z. Hou, B. Yu, Y. Qiao, X. Peng, and D. Tao, “Detecting human-object interaction via fabricated compositional learning,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognition, 2021, pp. 443–453.

[33] K. Kato, Y. Li, and A. Gupta, “Compositional learning for human object interaction,” in Proc. Eur. Conf. Comput. Vis., 2018, pp. 234–251.

[34] N.-G. Cho, S.-H. Park, J.-S. Park, U. Park, and S.-W. Lee, “Compositional interaction descriptor for human interaction recognition,” Neurocomputing, vol. 267, pp. 169–181, 2017.

[35] J. Materzynska, T. Xiao, R. Herzog, H. Xu, X. Wang, and T. Darrell, “Something else: Compositional action recognition with spatial-temporal interaction networks,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognition, 2020, pp. 1049–1059.

[36] W. Fan, C. Liang, and T. Wang, “Contrastive semantic disentanglement in latent space for generalized zero-shot learning,” Knowl.-Based Syst., vol. 257, 2022, Art. no. 109949.

[37] Z. Chen et al., “Semantics disentangling for generalized zero-shot learning,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2021, pp. 8712–8720.

[38] X. Li, Z. Xu, K. Wei, and C. Deng, “Generalized zero-shot learning via disentangled representation,” in Proc. AAAI Conf. Artif. Intell., 2021, pp. 1966–1974.

[39] B. Li, C. Han, T. Guo, and T. Zhao, “Disentangled features with direct sum decomposition for zero-shot learning,” Neurocomputing, vol. 426, pp. 216–226, 2021.

[40] Y. Geng et al., “Disentangled ontology embedding for zero-shot learning,” in Proc. 26th ACM SIGKDD Conf. Knowl. Discov. Data Mining, 2022, pp. 443–453.

[41] 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,” IEEE Trans. Neural. Netw. Learn. Syst., to be published, 2022, doi: 10.1109/TNNLS.2022.3176282.

[42] Z. Liu, Y. Li, L. Yao, J. McAuley, and S. Dixon, “Rethink, revisit, revise: A spiral reinforced self-reinforced network for zero-shot learning,” IEEE Trans. Neural. Netw. Learn. Syst., early access, May 27, 2022.

[43] Y. Li, Z. Liu, L. Yao, X. Wang, J. McAuley, and X. Chang, “An entropy-guided reinforced partial convolutional network for zero-shot learning,” IEEE Trans. Circuits Syst. Video Technol., vol. 32, no. 8, pp. 5175–5186, Aug. 2022.

[44] C. Li, G. Wang, B. Wang, X. Liang, Z. Li, and X. Chang, “DS-Net++: Dynamic weight slicing for efficient inference in CNNs and vision transformers,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 45, no. 4, pp. 4430–4446, Apr. 2023.

[45] Z. Yang, L. Zhu, Y. Wu, and Y. Yang, “Gated channel transformation for visual recognition,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognition, 2021, pp. 1383–1391.

[46] 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., 2020, pp. 21960–21980.

[47] A. Yu and K. Grauman, “Fine-grained visual comparisons with local learning,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognition, 2014, pp. 190–199.

[48] A. Yu and K. Grauman, “Semantic jitter: Dense supervision for visual comparisons via synthetic images,” in Proc. IEEE Int. Conf. Comput. Vis., 2017, pp. 5570–5579.

[49] P. Isola, J. J. Lim, and E. H. Adelson, “Discovering states and transformations in image collections,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognition, 2020, pp. 11 794–11 803.

[50] D. A. Hudson and C. D. Manning, “GQA: A new dataset for real-world visual reasoning and compositional question answering,” in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognition, 2019, pp. 6700–6709.

[51] G.-S. Xie et al., “Attentive region embedding network for zero-shot learning,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognition, 2019, pp. 192–199.

[52] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” in Proc. Adv. Neural Inf. Process. Syst., 2013, pp. 3111–3119.

[53] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, “Enriching word vectors with subword information,” Trans. Assoc. Comput. Linguistics, vol. 5, pp. 135–146, 2017.

[54] A. Yu and K. Grauman, “Semantic jitter: Dense supervision for visual comparisons via synthetic images,” in Proc. IEEE Int. Conf. Comput. Vis., 2017, pp. 5570–5579.

[55] 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., 2020, pp. 21960–21980.

[56] G.-S. Xie et al., “Attentive region embedding network for zero-shot learning,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognition, 2019, pp. 192–199.

[57] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” in Proc. Adv. Neural Inf. Process. Syst., 2013, pp. 3111–3119.

[58] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, “Enriching word vectors with subword information,” Trans. Assoc. Comput. Linguistics, vol. 5, pp. 135–146, 2017.

[59] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognition, 2016, pp. 770–778.

[60] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” 2014, arXiv:1412.6980.

[61] A. Krogh and J. Hertz, “A simple weight decay can improve generalization,” in Proc. Adv. Neural Inf. Process. Syst., 1991, pp. 950–957.

[62] A. Paszke et al., “PyTorch: An imperative style, high-performance deep learning library,” in Proc. Adv. Neural Inf. Process. Syst., 2019, pp. 8026–8037.
Zhe Liu received the PhD degree from the School of Computer Science and Engineering, and University of New South Wales (UNSW), Sydney, NSW, Australia. He is a postdoc with the University of New South Wales. His current research interests include zero-shot learning, generative network, data mining and their applications to computer vision, and human activity recognition.

Yun Li is currently a postdoc with CSIRO’s Data61, NSW, Australia. Her current research interests include zero-shot learning, transfer learning and their applications to computer vision, and genetic engineering.

Lina Yao (Senior Member, IEEE) is currently a senior principal research scientist and a science lead for Translational Machine Learning with CSIRO’s Data61 and conjoint professor with CSE, UNSW. Her current research interests include data mining and machine learning with applications to Internet of Things, information filtering and recommending, human activity recognition, and brain-computer interface.

Xiaojun Chang (Senior Member, IEEE) is a professor with Australian Artificial Intelligence Institute, University of Technology Sydney. He is also a visiting professor with the Department of Computer Vision, Mohamed bin Zayed University of Artificial Intelligence (MBZUAI). Before joining UTS, he was an associate professor with the School of Computing Technologies, RMIT University, Australia. After graduation, he subsequently worked as a postdoc research fellow with School of Computer Science, Carnegie Mellon University, lecturer and senior lecturer with the Faculty of Information Technology, Monash University, Australia. He has spent most of his time working on exploring multiple signals (visual, acoustic, textual) for automatic content analysis in unconstrained or surveillance videos. He has achieved top performances in various international competitions, such as TRECVID MED, TRECVID SIN, and TRECVID AVS.

Wei Fang (Member, IEEE) received the PhD degree in information technology and engineering of light industry from Jiangnan University, Wuxi, China, in 2008. He is a professor of computer science with Jiangnan University. He was a visiting scholar with the CEACIA, University of Birmingham, Birmingham, U.K. from 2013 to 2014. He has published more than 50 scientific papers in journals and international conferences.

Xiaojun Wu (Member, IEEE) received the BSc degree in mathematics from Nanjing Normal University, Nanjing, China, in 1991, and the MS and PhD degrees in pattern recognition and intelligent system from Nanjing University of Science and Technology, Nanjing, in 1996 and 2002, respectively. From 1996 to 2006, he taught with the School of Electronics and Information, Jiangsu University of Science and Technology, where he was promoted to professor. He has been with the School of Information Engineering, Jiangnan University since 2006, where he is a professor of pattern recognition and computational intelligence. His current research interests include pattern recognition, computer vision, and computational intelligence. He was a fellow of the International Institute for Software Technology, United Nations University, from 1999 to 2000. He was a recipient of the Most Outstanding Postgraduate Award from Nanjing University of Science and Technology.

Abdulmotalab El Saddik (Fellow, IEEE) is currently the acting chair with Mohamed bin Zayed University of Artificial Intelligence and a distinguished professor with the School of Electrical Engineering and Computer Science, University of Ottawa. He is also an internationally recognized scholar who has made strong contributions to the knowledge and understanding of multimedia computing, communications, and applications, particularly in the area of digital twins. He is a fellow of the Royal Society of Canada, the Canadian Academy of Engineering, and the Engineering Institute of Canada. He is an ACM Distinguished Scientist and has received several awards, including the Friedrich Wilhelm Bessel Award from the German Humboldt Foundation and the IEEE Instrumentation and Measurement Society Technical Achievement Award. He also received IEEE Canada C. C. Gotlieb (Computer) Medal and A. G. L. McNaughton Gold Medal for important contributions to the field of computer engineering and science and the 2021 IEEE TCSC Achievement Award for Excellence in Scalable Computing.