Semantic Graph-enhanced Visual Network for Zero-shot Learning

Yang Hu[1,2], Guihua Wen[1], Adriane Chapman[2], Pei Yang[1], Mingnan Luo[1], Yingxue Xu[1], Dan Dai[1], and Wendy Hall[2]

[1] South China University of China
Guangzhou 510006, China
[2] University of Southampton
Southampton SO171BJ, UK

Abstract. Zero-shot learning uses semantic attributes to connect the search space of unseen objects. In recent years, although the deep convolutional network brings powerful visual modelling capabilities to the ZSL task, its visual features have serious pattern inertia and lack of representation of semantic relationships, which leads to severe bias and ambiguity. In response to this, we propose the Semantic Graph-enhanced Visual Network to conduct graph modelling of visual features mapped to semantic attributes by using a knowledge graph, it contains several novel designs: 1. Refining the visual features with graph convolutional networks on the semantic graph; 2. Using attribute word embedding as a target for a semantic regression; and 3. Fusion and supplementing the intermediate visual features refined by semantic graph into visual embedding. By promoting the semantic linkage modelling of visual features, our method outperforms state-of-the-art approaches on multiple representative ZSL datasets: AwA2, CUB, and SUN.

Keywords: Zero-shot learning; graph convolutional network; semantic knowledge graph; attribute word embedding

1 Introduction

Zero-shot learning (ZSL) presents a new paradigm for image recognition, whose goal is to train models to recognize categories that have never been seen before [22]. Because many instances in datasets in the real world do not have associated classes, ZSL is required to make computer vision (CV) more generalized [20,47,17,16,30]. Unlike typical supervised learning, in which the unlabelled is a part of the samples. In ZSL, the unlabelled are the entire categories (unseen), and the only clue that connects the seen and unseen classes is the attribute assigned to each class. In recent years, architectures using deep convolutional networks (CNN) are used as the embedding subnet for ZSL [38,49,25,16]. CNN based ZSL models are usually pre-trained by ImageNet [34] to earn the initial visual perception ability [38,49,52]. However, the models only provide mappings between classes and observed patterns; knowledge about relationships within the attributes of seen and unseen classes are not utilized.
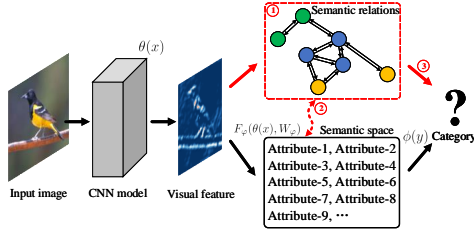
Fig. 1. Illustrating the strategy of semantic graph-enhanced visual embedding. The red lines and dashed box represent the modelling of implicit semantic links in visual features and help visual-semantic bridging complete the category mapping.

In existing ZSL methods, both general inductive setting methods [51,33,52] and transductive setting methods [21,38,42] which even access to unseen samples, the visual image modelling and attribute mapping modelling are separate. For example, in a ZSL task, the input image is denoted as x, the visual embedding function formed by CNN is $\theta(\cdot)$. Separated from this, the semantic embeddings $\phi(y)$ indicate the attribute distribution of each class. $F(x, y, W_\varphi) = F_\varphi(\theta(x), W_\varphi)\phi(y)$ is the score function of classification. The only connection between the visual and the semantic embeddings is the visual-semantic bridging $F_\varphi(\cdot, W_\varphi)$, which is usually constructed by few fully-connected layers (FC). Facing visual embedding $\theta(\cdot)$ with strong pattern inertia, $F_\varphi(\cdot, W_\varphi)$ bears an excessive modelling pressure and hard to reverse prediction bias. Although the latent attributes mechanism reduces the skew caused by the attribute inertia [25,27], the existing ZSL models still ignore the implicit semantic linkages in visual features. The components outlined in black in Figure 1 show the current state of the art in how ZSL uses attributes.

In order to address the lack of knowledge about relationships within the attributes, we must use knowledge of the attribute connections provided by their semantic relationships, as shown in red in Figure 1. To do this, we propose the **Semantic Graph-enhanced Visual** (SGV) network, which consists of 3 key points: 1. In order to use the relationships amongst attributes found in the semantic knowledge graph, we use graph convolutional network (GCN) [19] to remodel the intermediate features of the visual model and feedback the semantic information to the CNN. Our GCN's running structure is based on the attribute co-occurrence knowledge graph; 2. In order to provide better sets of related attributes for the model, we create a self-consistent system with GCN modelling by creating attribute word vectors to achieve semantic regression. 3. Finally, in order to integrate this new semantic-derived information into the classic ZSL model, we skip-connect the graph modeled visual features to the visual embedding $\theta(x)$. The purpose is to fuse the graph enhanced features to visual embedding and help the loss back-propagation from semantic embedding $\phi(y)$.

The contributions of our study can be summarised as:

1. We propose the **Semantic Graph-enhanced Visual** network, in which, GCN remodels the CNN visual features with an attribute knowledge graph information, and provides semantic relations to a deep visual model.

2. We use the attribute word vectors as the regression output of the GCN pipeline, which offers supervision for semantic graph modelling and completes a self-consistent system.

3. We combine the graph enhanced features into visual embedding to enhance the semantic relational representation of visual embedding.

4. We evaluate the proposed SGV network and supporting techniques. The proposed method achieves state-of-the-art performance on ZSL datasets: AwA2 [46], CUB [41], and SUN [32]. We perform ablation studies and visualization analysis to validate the effectiveness of our designs.

2 Related Works

2.1 Zero-shot Learning

Zero-shot learning (ZSL) uses semantic attributes to connect the search space of unseen objects. The original ZSL uses a two-stage method; the first stage predicts the attribute while the second stage maps the attribute to the class, based on direct attribute prediction (DAP) [22]. The prediction of ZSL relies on a shared semantic space which can be user-defined attribute annotations [11] or unsupervised word embeddings [28].

In order to both remove the gap between attribute predictions to the final category and correct assumptions that attributes are independent of each other, ESZSL [33] and SCoRe [29] propose to use a multi-layer neural network to train the visual-semantic bridge. SAE [20] introduces external word embeddings as an assistant. However, the "*domain shift*" [13] still causes severe category bias, especially in generalized ZSL setting [4], which mix the seen and unseen classes in a search space. [16,48] add regulation in the loss function. AEZSL [31] generates embedding spaces from multiple perspectives and PREN [49] uses the ensemble method to fuse them. The adversarial learning strategy is also applied in ZSL to generate more vivid embedding spaces [52,30,47]. The transductive ZSL methods [38,49,42] even trains some part of unlabeled unseen class samples to make up for the natural lacking of inference clues in visual-semantic modelling. Moreover, LDF [25] suggest that human-defined attributes cannot fully depict semantic information and propose the latent attribute mechanism to automatically learn the pseudo-attributes. LFGAA [27] adds attention values for both human-defined and pseudo attributes.

Instead, we argue that another vital reason for the "domain shift" is that the connection between visual features and semantic space is too weak. We gain the obvious difference between the existing ZSL methods and the SGV: 1. The embedding subnet in previous works focuses on modelling the visual features and ignores the implicit semantic linkages in deep visual features; 2. The SGV network introduces the unbiased knowledge graph to supervise the visual modelling and uses the GCN to learn the cognition of semantic relations.

Some methods use external knowledge to assist ZSL models [20,29,45,17]. Among them, SEKG [45] and DGP [17] are closest to our study. However, they are still extremely different from SGV: 1. SEKG and DGP rely on the topology of WordNet. Thus the applicable datasets are limited; 2. Although they also use a GCN for semantic modelling, it is insufficient for deep visual representation. 3. They lack the ability to feed the outcomes of semantic graph learning back to the visual features.

2.2   Graph Neural Network for Computer Vision

Graph neural networks (GNNs) are widely applied in computer vision due to their excellent modelling capabilities for non-Euclidean relations [3,19]. The benefit of visual modelling applications include: various recognition tasks [8,7], object detection [39] and segmentation [50], as well as visual point cloud data modelling [23]. Researchers use a GNN to model the links between category labels [9,5] or integrate external knowledge [8]. Graph learning also enhances the representation of the connections between visual objects and semantic entities and refines the boundaries of detection and segmentation [50]. Visual-semantic interaction tasks like visual question & answer (VQA) and image caption are emphasised application areas of a GNN [24,43], though which the logic relations between visual features are modeled, which gives the model semantic reasoning ability and promotes the info-transmission of visual signals to semantic.

A graph convolution [19] is the most common method of graph learning. In addition the above studies can confirms that implicit semantic linkages do exist in visual features, graph modelling is a powerful means of loading structured knowledge into deep models.

The above methods are significantly different from the proposed SGV network: 1. In the existing studies, a GNN models the outcomes of a CNN or external structured knowledge without modelling intermediate visual features; 2. The previous GNN-based models lack the ability to feed the graph modelled features back the visual representation. By contrast, the SGV method uses a GCN to reconstruct and readjust deep visual representations, which enhances the ability to model semantic relationships in visual models while retaining the ability to express them visually.

3   Semantic Graph-enhanced Visual Network

3.1   Problem Formulation and Notations

The ZSL task is formulated as follows: there is a seen dataset $\mathcal{S} = \{(x_i^s, y_i^s)\}_{i=1}^{n^s}$ which contains n^s samples for training, where x_i^s denotes the i-th image and $y_i^s \in \mathcal{Y}^S$ is the category label of it. There is an another unseen dataset $\mathcal{U} = \{(x_i^u, y_i^u)\}_{i=1}^{n^u}$ with similar form. The seen and unseen category sets $\mathcal{Y}^S$ and $\mathcal{Y}^U$ obey the following constraints: $\mathcal{Y}^S \cap \mathcal{Y}^U = \emptyset, \mathcal{Y}^S \cup \mathcal{Y}^U = \mathcal{Y}$ where $\mathcal{Y}$ is the total category set. $\mathcal{Y}^S$ and $\mathcal{Y}^U$ share a semantic attribute space: $\forall y_i \; \exists < a_1, \ldots, a_m >$

Fig. 2. Illustration of attribute link construction. The gray-dotted lines are the co-occurrence relations with low $\mathcal{PMI}$ value, and the black lines indicate a high $\mathcal{PMI}$ value between attributes. There is no separate sub-graph in our knowledge graph, and '?' represents another attribute vertex that connects to the appeared attributes.

as the only bridge between them and $y_i \in \mathcal{Y}$, and m is the number of attributes. The goal of conventional ZSL (CZSL) is to learn the classifier with the search space of unseen classes $\mathcal{Y}^{\mathcal{U}}$, for more challenging generalized ZSL (GZSL), the search space of expected classifier is $\mathcal{Y}$.

3.2 Overview Framework

In order to achieve both the goal of CZSL and GZSL, the following steps must be taken: 1. **Construction of Semantic Knowledge Graph:** Construct the semantic knowledge graph with co-occurrence relations of attributes. 2. **Knowledge Injection:** Transform the knowledge graph into information that can be associated with visual features using at GCN, then inject information from the knowledge graph into the CNN. 3. **Auxiliary Supervision:** Use word vectors to regulate the graph modelling of attribute semantic relationships. 4. **Hybrid Embedding:** Merge the semantic topic features modeled by the GCN into the final visual embedding.

Construction of Semantic Knowledge Graph Instead of applying Word-Net as a prior knowledge graph as done in [45,17], we construct the semantic knowledge graph based on attribute co-occurrence relations. The attributes with co-occurrence relation will have a high probability of existing in the same visual image.

A semantic knowledge graph, $\mathcal{G}_{att} = (\mathcal{V}, \mathcal{E})$, contains vertices $\mathcal{V} = \{v_1, v_2, \ldots, v_m\}$ and edges $\mathcal{E})$ between them. We utilize a symmetric matrix to encode edges $[[l_{i,j}]]$ where $l_{i,j} = 1$ means there is a linkage between vertices v_i and v_j, else not. The point-wise mutual information (PMI) [2] is used to determine the connection between vertices, as follows:

$$\mathcal{PMI}(v_i, v_j) = \mathcal{N}\left(\log \frac{p(v_i, v_j)}{p(v_i)\,p(v_j)}\right), \tag{1}$$

where $\mathcal{PMI}(v_i, v_j)$ is the PMI between attributes v_i and v_j, $p(v)$ is the occurrence probability of attribute v, and $p(v_i, v_j)$ is the co-occurrence probability of
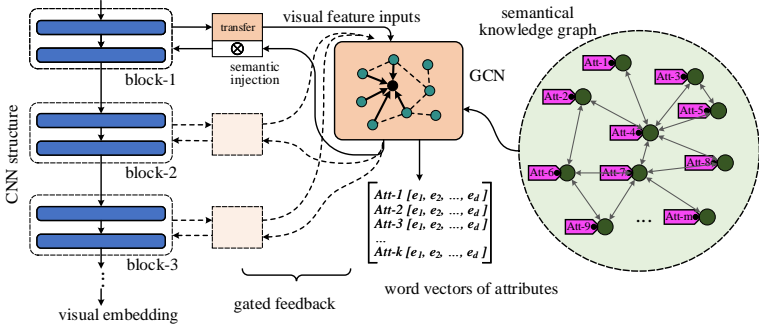
Fig. 3. Overview of SGV network. The leftmost area shows the CNN blocks, the middle shows the transfer and graph modelling module, and the rightmost area is the semantic knowledge graph, the vertexes of which refer to the attributes.

v_i and v_j. $\mathcal{N}$ denotes the normalization function within $[0, 1]$. Figure 2 illustrates the construction strategy of the semantic knowledge graph, and the link is built between the vertices with a PMI of higher than the threshold δ.

Knowledge Injection Now that we know the co-occurrence relations of semantic attributes, we can use this knowledge information to optimize the visual model. The efficacy of the SGV network is based on a premise that there are implicit semantic linkages in visual features, and this has been found in other CV studies [44]. Our approach will provide the CNN model both the cognitive ability of visual and semantic relationships. This is different to other ZSL methods using attention mechanism to generate attentive regions [47] or adjust the weights of human-defined and latent attributes [27]. As Figure 3 shows, in our SGV network, the attention mechanism is a tool to inject the semantic knowledge to visual features. Given the feature maps X of any block in the CNN-based visual model, semantic graph modelling is defined as:

$$f_{\mathcal{G}} = F_{\mathcal{G}} \left(< F_{in} \left(X, W_{in} \right), \mathcal{G}_{att} >, W_{\mathcal{G}} \right), \tag{2}$$

where the transform function $F_{in}(\cdot, W_{in})$ converts X to the suitable input shape of the GCN, and W_{in} is the transform weights. $F_{\mathcal{G}}(< \cdot, \mathcal{G}_{att} >, W_{\mathcal{G}})$ is the SGV block parameterized by $W_{\mathcal{G}}$, which contains two GCN layers. $f_{\mathcal{G}}$ is the graph enhanced features of SGV block.

The graph convolution is defined as $H^{(i+1)} = \sigma \left(D^{-1} G_a H^{(i)} W_{\mathcal{G}}^{(i)} \right)$ [19], where $H^{(i)}$ and $H^{(i+1)}$ are the layers of GCN, D and G_a are the degree and adjacency matrices of the preset graph $\mathcal{G}_{att}$, σ refers to an activation function.

Next, we inject the graph enhanced information $f_{\mathcal{G}}$ to visual features X, as:

$$\tilde{X} = \sigma_{out} \left(f_{\mathcal{G}}, W_{out} \right) \otimes X, \tag{3}$$
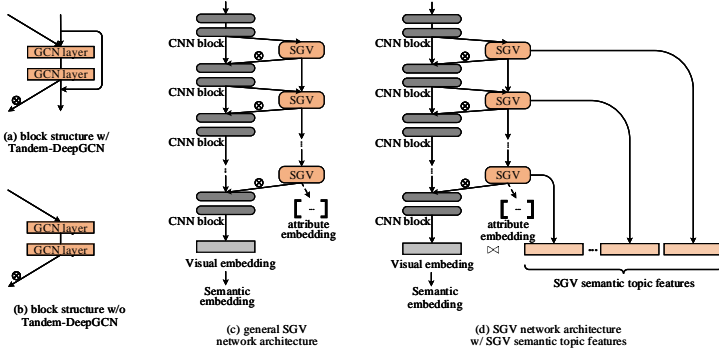
Fig. 4. Detailed architecture of the SGV network. (a-b) Different SVG blocks with or without Tandem-DeepGCN; (c) The general SGV network structure; (d) The SGV network with the semantic topic features (STF) that fused into final visual embedding.

where $\tilde{X}$ is the new visual features enhanced with semantic graph modelling, and $\otimes$ is the element-wise multiplication. $\sigma_{out}(\cdot, W_{out})$ is the activation and transform function with weights W_{out}.

Auxiliary Supervision Now that we have the knowledge injection method, we need to use regulation to optimize the semantic graph modelling with attribute word vectors, which is different from previous work that learns the semantic word embeddings in ZSL [20,31]. We set the target output of the GCN pipeline to the attribute word vectors $\mathcal{A} = [a_1, a_2, \ldots, a_k]$ of the corresponding category y, where $a_i \in \mathbb{R}^d$ is the word vector of one attribute and k is the number of attributes belong to category y, d is the dimension of word embedding. We consider each category y as a bag of words $\{Att_1, Att_2, \ldots, Att_k\}$, including its corresponding semantic attributes. The training corpora of attribute word vectors is all category word bags.

To feed-forward multiple layers of semantic relation information, we connect all SVG blocks to form a Tandem-DeepGCN pipeline. As shown in Figure 4(c), each SGV block receives both the previous GCN features and the CNN visual features as inputs. Given the previous GCN features $f_{\mathcal{G}}^{(i-1)}$, we obtain:

$$f_{\mathcal{G}}^{(i)} = F_{\mathcal{G}}^{(i)}\left(< F_{in}(X, W_{in}) \bowtie F_{sq}(f_{\mathcal{G}}^{(i-1)}), \mathcal{G}_{att} >, W_{\mathcal{G}}^{(i)}\right), \qquad (4)$$

where $F_{sq}(\cdot)$ is a transform function squeezing the dimension of the GCN features, and $\bowtie$ is concatenation. To solve the gradient diffusion on the GCN pipeline, we adopt the residual GCN structure $H^{(i+1)} = F_{\mathcal{G}}(H^{(i)}, W_{\mathcal{G}}^{(i)}) + H^{(i)}$ [23]. So, we implement deep modelling on the semantic knowledge graph using the Tandem-DeepGCN pipeline, as shown in Figure 4(c). And we optimize this deep graph modelling by using the regression of attribute word vectors.

Hybrid Embedding To further enhance the final visual embedding, we perform one last step. As illustrated in Figure 4(d), inspired by the pyramid structure [26], we merge the enhanced visual-semantic features $f_{\mathcal{G}}$ into the final visual embedding $\theta(x)$, i.e. **Semantic Topic Features (STF)**. Its advantages are: 1. Help the loss back-propagation from ZSL semantic embedding $\phi(y)$ to the SGV blocks; 2. Provide more layers of semantic graph enhanced information for the latent attributes [25,27]. As follows:

$$\theta(x)^+ = \theta(x) \bowtie \hat{f_{\mathcal{G}}}^{(1)} \bowtie \hat{f_{\mathcal{G}}}^{(2)} \bowtie \cdots \bowtie \hat{f_{\mathcal{G}}}^{(L)}, \tag{5}$$

where $\theta(x)^+$ is the integrated visual embedding, and $\hat{f_{\mathcal{G}}}^{(i)}$ refers to the squeezed STF and is calculated as $\hat{f_{\mathcal{G}}}^{(i)} = \frac{1}{m} \sum_{v=1}^{m} \left(F_{sq}(f_{\mathcal{G}}^{(i)})[v] \right)$. Moreover, m is the number of attribute vertices, and $F_{sq}(\cdot)$ squeezes the SGV features to avoid $\theta(x)^+$ from becoming too wide. Because $\theta(x)$ is obtained from the global average pooling of all feature maps, we average the STF along the vertical axis.

4   Optimization

The weights in neural network of the proposed SGV network described above is optimised using a multi-task system. Given the semantic prediction $\varphi(x) = F_\varphi(\theta(x), W_\varphi)$, which is calculated using visual-semantic bridging, we use the softmax cross-entropy loss:

$$\mathcal{L}_A = -\frac{1}{n} \sum_{i}^{n} \log \frac{\exp\left(\varphi(x_i)^\top \phi(y_i)\right)}{\sum_{y^s \in \mathcal{Y}^S} \exp\left(\varphi(x_i)^\top \phi(y^s)\right)}. \tag{6}$$

We use the mean-square error as the loss of auxiliary supervision based on the attribute word vectors. Note that for each training pair (x_i, y_i), except for the k attribute word vectors belonging to category y_i, the ground-truth of the remaining $m - k$ attribute word vectors is set to 0:

$$\mathcal{L}_W = \frac{1}{n} \sum_{i}^{n} \sum_{v}^{m} \left\| F_{sq}\left(f_{\mathcal{G}}^{(L)}(x_i)\right)[v] - \mathrm{a}_v(y_i) \right\|^2. \tag{7}$$

We also follow the same mechanism of latent attributes in [25,27], which applies triplet loss [35] to learn discriminative latent category features:

$$\mathcal{L}_{LT} = -\frac{1}{n} \sum_{i}^{n} \left[\|\zeta(x_i) - \zeta(x_j)\|^2 - \|\zeta(x_i) - \zeta(x_r)\|^2 + \alpha \right]_+, \tag{8}$$

where $\zeta(x)$ refers to the visual-latent prediction, x_i and x_j are images from the same class and x_r is from a different class. $[\cdot]_+$ is equivalent to $\max(0, \cdot)$. α is the margin of triplet loss and is 1.0 in all experiments.

We combine the above learning targets with the balancing factors β and γ:

$$\mathcal{L} = \mathcal{L}_A + \beta\mathcal{L}_{LT} + \gamma\mathcal{L}_W \tag{9}$$

5   Experiments and Analysis

5.1   Experimental Setup

Datasets We conduct the experiments on three representative ZSL datasets: Animals with Attribute 2 [46] (AwA2), Caltech-UCSD Birds 200-2011 [41] (CUB) and Scene Classification Database [32] (SUN). AwA2 is a coarse-grained dataset containing 37,322 images from 50 animal classes with 85 attributes. CUB is a fine-grained dataset consisting of 11,788 images from 200 different bird species with 312 attributes. SUN is another fine-grained dataset that including 14,340 images from 717 different scenes providing 102 attributes. The zero-shot splits are adopted as 40/10, 150/50, and 645/72 on AwA2, CUB, and SUN, respectively, for two split strategies: standard split and proposed split [46].

Evaluation metrics The average per-class top-1 accuracy $\mathcal{ACC}$ is used as the primary metric. The experiments include both conventional and generalised ZSL settings [46]. In the CZSL setting, all test samples come from the unseen classes ($y \in \text{Set}\langle \mathcal{Y}^U \rangle$). In the GZSL setting, the test samples come from both seen and unseen classes ($y \in \text{Set}\langle \mathcal{Y}^S \rangle \cup \text{Set}\langle \mathcal{Y}^U \rangle$), and we report the accuracy $\mathcal{ACC}^S$, $\mathcal{ACC}^U$ of seen and unseen test samples and their harmonic mean $\mathcal{H} = \frac{2 \times \mathcal{ACC}^S \times \mathcal{ACC}^U}{\mathcal{ACC}^S + \mathcal{ACC}^U}$.

Implementations The backbone network ResNet [14] is applied to initialise the Embedding Subnet using the pre-trained weights on ImageNet-1k dataset [10]. Moreover, we test ResNet [14] with 18, 50, and 101 layers as a backbone to demonstrate that the proposed SGV block also works well on the smaller Embedding Subnet. This is the first study to conduct such a comparison.

For other hyper-parameters in the SGV network, the In and Out transform functions $F_{in}(\cdot)$ and $\sigma(\cdot)$ in Eqs. 2 and 3 are implemented with a FC mapping, and the activation function of $\sigma(\cdot)$ is Sigmoid. Each SGV block contains two GCN layers with the dimension of 64 and $h \times w$, where (h, w) is the shape of the visual feature maps. The FC mapping $F_{sq}(\cdot)$ squeezes the features dimension to 10. The threshold δ and balancing factors β, γ are set to 0.75, 1.0, and 0.5, respectively. We apply the Adam optimizer [18] throughout all experiments. Unless otherwise specified, our SGV model uses the attribute word embedding sub-task and STF by default.

5.2   Conventional Comparison

Overall performance. CZSL mainly tests the recognition ability of the model for unseen samples. We compare several state-of-the-art transductive ZSL methods [21,38,49,27] and inductive ZSL approaches [22,51,12,33,1,15,47,16,52]. In addition to an ordinary SGV network, we also add the models of SGV^{LT} (with the latent attributes in [25]), $\text{SGV} + \text{T}$ (using the transductive method in [38]), and $\text{SGV}^{LT} + \text{SA}$ (using the transductive method in [27]) into the competition.

Table 1 shows that the proposed SGV network outperforms the existing ZSL methods with both inductive and transductive strategies on CZSL. In most cases, our SGV can achieve the best results. Even with fewer layers, the SGV remains

Table 1. Comparisons under the CZSL setting (%). For each dataset, the best, the second-best, and the third-best performances are marked in **bold**, blue, and green font respectively for both the inductive and transductive methods. For SGV-18, SGV-50, and SGV-101, the visual embedding subnet is ResNet with 18, 50, and 101 layers. Both standard split (SS) and proposed split (PS) are considered. The notations: $\mathcal{I}$ denotes inductive ZSL methods, and $\mathcal{T}$ denotes transductive ZSL methods.

|	Method	AwA2		CUB		SUN	
		SS	PS	SS	PS	SS	PS
$\mathcal{I}$	DAP [22]	58.7	46.1	37.5	40.0	38.9	39.9
	SSE [51]	67.5	61.0	43.7	43.9	25.4	54.5
	DEVISE [12]	68.6	59.7	53.2	52.0	57.5	56.5
	ESZSL [33]	75.6	55.1	43.7	53.9	57.3	54.5
	ALE [1]	80.3	62.5	53.2	54.9	59.1	58.1
	CDL [15]	79.5	67.9	54.5	54.5	61.3	63.6
	AREN [47]	**86.7**	67.9	**70.7**	71.8	61.7	60.6
	LFGAA [27]	84.3	68.1	67.6	67.6	62.0	61.5
	TCN [16]	70.3	71.2	–	59.5	–	61.5
	SGMA [52]	83.5	68.8	70.5	71.0	–	–
	SGV-18 (ours)	77.3	67.5	68.1	67.2	59.2	59.0
	SGV-50 (ours)	82.6	71.4	69.2	70.3	61.6	61.7
	SGV-101 (ours)	85.8	73.2	70.6	72.2	64.3	63.8
	SGV^{LT}-101 (ours)	88.1	74.0	71.4	72.8	64.2	64.8
$\mathcal{T}$	SE-ZSL [21]	80.8	69.2	60.3	59.6	64.5	63.4
	QFSL [38]	84.8	79.7	69.7	72.1	61.7	58.3
	PREN [49]	95.7	74.1	66.9	66.4	63.3	62.9
	LFGAA + SA [27]	94.4	84.8	79.7	78.9	64.0	66.2
	SGV-50 + T (ours)	91.0	80.5	74.2	76.0	60.5	62.6
	SGV-101 + T (ours)	93.2	83.7	77.6	79.4	64.0	63.8
	SGV^{LT}-101 + SA (ours)	96.3	86.8	81.0	82.3	65.0	67.9

extremely competitive. SGV-50 can outperform most state-of-the-art methods. The performance of SGV-18 can also approach that of other state-of-the-art models that are usually based on a much larger backbone.

Our SGV achieves an improvement $2.8\% \sim 5.2\%$ over LFGAA, which also applies an attention mechanism. The improvement by SGV is greater than 2% in the transductive setting. The comparative methods frequently only show an excellent performance on one dataset, while the SGV can achieve the best on all datasets. Moreover, we found that SGV performs well on fine-grained classification datasets such as CUB and SUN, which is most evident on the CUB datasets with detailed attribute descriptions. This indicates that the SGV has strong cross-modal semantic modelling capabilities on fine-grained visual features and clearly described attributes.

Analysis of the SGV vs other attention mechanisms. As Table 2 shows, we compare the models with and without the SGV to validate its effectiveness on ZSL. We chose the inductive version of QFSL [38] (denoted as B-Net$^{w/o\ SGV}$) as the baseline. To better show the benefit of SGV, we introduce two baselines Att-Net and SCA-Net with spatial attention and both spatial and channel-wise attention mechanisms [6]. We also analyse the effects of configuring the SGV block in each CNN block, each stage (when the size of the receptive field changes), or the last CNN block. Although the ordinary attention mechanism can improve the baseline slightly, it cannot compare the ability of graph

Table 2. Ablation results with or without SGV on proposed split (PS) (%). For $SGV^{(last\ block)}$, $SGV^{(each\ stage)}$, and $SGV^{(each\ block)}$, the SGV block is added to each CNN block, each stage, and the last CNN block of the embedding subnet.

| Option	Method	AwA2	CUB	SUN
w/o SGV	B-Net$^{w/o\ SGV}$	62.7	59.0	56.5
	Att-Net [6]	63.9	64.2	57.7
	SCA-Net [6]	63.3	65.5	59.0
w/ SGV	$SGV^{(last\ block)}$	72.8	70.4	61.9
	$SGV^{(each\ stage)}$	**73.2**	71.8	62.9
	$SGV^{(each\ block)}$	72.0	**72.2**	**63.8**
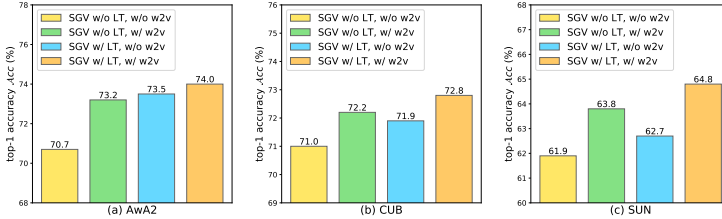
Fig. 5. Comparisons on AwA2, CUB, and SUN datasets with proposed split (PS) (%). We apply the inductive setting and set the latent attribute mechanism [25] and attribute word vector sub-task as options.

modelling for a semantic representation, and the SGV shows overwhelming advantages. Moreover, the granularity of the dataset decides requirements for SVG blocks. On AwA2, fewer SGV blocks achieve the best results, indicating that semantic attributes are relatively obscure in coarse-grained visual features.

Analysis of the attribute word vector sub-task. We validate the improvement from applying the attribute word vector sub-task. As illustrated in Figure 5, we can see that the regression of the semantic graph modelling by an attribute word vector sub-task can significantly improve the performance of visual-semantic modelling. Particularly on fine-grained image recognition datasets, like CUB and SUN, the benefits of word embedding regression are higher than the latent attributes. We analyze the reason is that the feedback of fine-grained visual features on the attributes is more prominent, and the association of a fine-grained visual semantic is stronger.

Moreover, when the attribute word vector sub-task and latent attributes are used at the same time, the classification accuracy reaches its peak, indicating that the regression of semantic graph modelling to attribute word vectors supporting ZSL model generates more accurate latent attributes.

Analysis of using STF. We conduct another ablation study to validate the effect of STFs and set the mechanism of the latent attribute as an option to test the collaborative performance of STF and the latent attribute mechanisms [25,27]. Figure 6 shows that, as STF concatenates more semantic features into a visual embedding, the performance of the model is continuously improved,
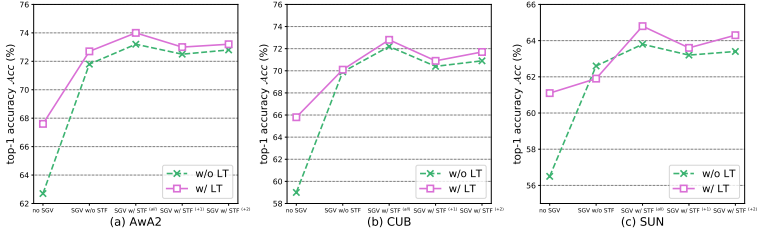
Fig. 6. Performance of different methods on AwA2, CUB, and SUN datasets with proposed split (PS) (%) in the inductive setting of CZSL tasks. Here, **no SGV** indicates baseline model with inductive ResNet QFSL [38], and different settings of STF are compared. Moreover, $\mathbf{STF}^{(all)}$ denotes that the semantic features from all SGV blocks are merged with the visual embedding, and $\mathbf{STF}^{(+1)}$ and $\mathbf{STF}^{(+2)}$ indicate that only the last one or two SGV blocks are considered.

Table 3. Comparisons in the GZSL setting (%). For each dataset, the best, the second best, and the third best results are marked in **bold**, **blue**, and **green** fonts respectively.

Method	AwA2			CUB			SUN		
	$\mathcal{ACC}^S$	$\mathcal{ACC}^U$	$\mathcal{H}$	$\mathcal{ACC}^S$	$\mathcal{ACC}^U$	$\mathcal{H}$	$\mathcal{ACC}^S$	$\mathcal{ACC}^U$	$\mathcal{H}$
DEVISE [12]	90.5	10.0	18.0	70.9	11.5	19.8	**43.3**	7.9	13.4
ESZSL [33]	77.8	5.9	11.0	63.8	12.6	21.0	27.9	11.0	15.8
CMT* [37]	89.0	8.7	15.9	60.1	4.7	8.7	28.0	8.7	13.3
CDL [15]	73.9	29.3	41.9	55.2	23.5	32.9	34.7	21.5	26.5
AREN [47]	79.1	54.7	64.7	69.0	**63.2**	**66.0**	**40.3**	**32.3**	35.9
TCN [16]	65.8	**61.2**	63.4	52.0	52.6	37.3	37.3	31.2	34.0
SGMA [52]	87.1	37.6	52.5	71.3	36.7	48.5	–	–	–
PREN [49]	88.6	32.4	47.4	55.8	35.2	27.2	35.4	27.2	30.8
LFGAA [27]	**93.4**	27.0	41.9	**80.9**	36.2	50.0	**40.0**	18.5	25.3
LFGAA + SA [27]	90.3	50.0	64.4	**79.6**	43.4	56.2	34.9	20.8	26.1
SGV-50 + T (ours)	90.8	**65.4**	**76.0**	69.2	**68.3**	**68.7**	36.9	**50.6**	**42.7**
SGV-101 + T (ours)	**94.0**	69.2	**79.7**	74.8	**73.1**	**73.9**	38.3	**55.2**	**45.2**
SGV^{LT}-101 + SA (ours)	**93.1**	57.1	**70.8**	**80.6**	53.2	64.1	**43.3**	31.4	**36.4**

and better results are obtained when we introduce latent attributes and STF concurrently. We can conclude that STF not only complements the visual embedding, it also boosts the modelling of the latent attributes.

5.3   Generalized Comparison

GZSL test model's ability to recognize samples in seen and unseen mixed search space. We verify the proposed SGV network in the GZSL task. As listed in Table 3, we can note that, except for a few classification results for seen classes, our proposed method outperforms other comparison methods on all datasets. SGV^{LT}-101 + SA improve over the state-of-the-art LFGAA + SA [27] by a significant margin of $6.4\% \sim 20.3\%$ on the AwA2, CUB, and SUN datasets. With a more straightforward application of unseen class samples (as the transductive setting in [38]), SGV-101 + T achieves $9.3\% \sim 17.7\%$ improvement over the comparison methods. Although the forced reversal of unseen class weights [38]
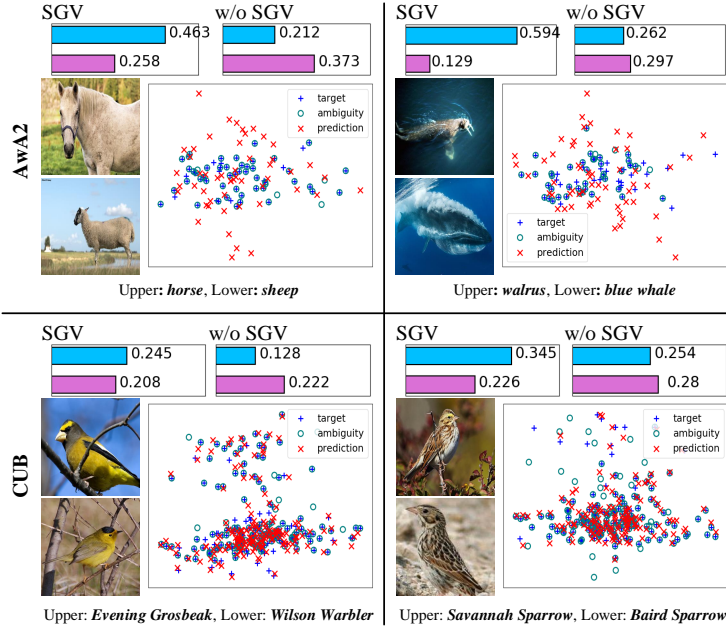
Fig. 7. Visualisation of the attribute word vector outcomes in CZSL task on AwA2 and CUB. The left side of each sub-figure are example images, the upper part is the input image, and the lower part is from the ambiguity class. The top of each sub-figure is the classification probabilities of the compared methods, the **blue** bar refers to the target class, and **pink** indicates the ambiguity class. The symbols '+', 'o', 'x' in the scatter represent the word vector projections of the target class, ambiguity class, and sub-output of the SGV, respectively. The PCA dimensionality reduction [40] is used.

will reduce the seen class accuracy on extremely biased datasets such as SUN, the SGV-101 + T and SGV-50 + T (with only a 50 layer backbone) still achieve the most competitive scores on the harmonic metric. Moreover, although many existing state-of-the-art methods use unseen images or semantics during training, they do not achieve the best result on all datasets, unlike the SGV network.

5.4   Visualization and Disambiguation Analysis

To gain more insightful evidence into the effectiveness of the SGV network, we conduct visualisation experiments to show the performance of the SGV in semantic graph modelling and visual feature attention, and analyze the disambiguation of the SGV network.

We apply inductive SGV^{LT}-101 and LFGAA to compare the models with or without the SGV, as shown by Figure 7. Note that the target class and the ambiguity class share numerous attributes, and thus we can see that many symbols similar to '⊕' indicate the coincidence of the attributes. For each dataset, the left and right sub-figures respectively indicate strong and slightly weak ambiguities.
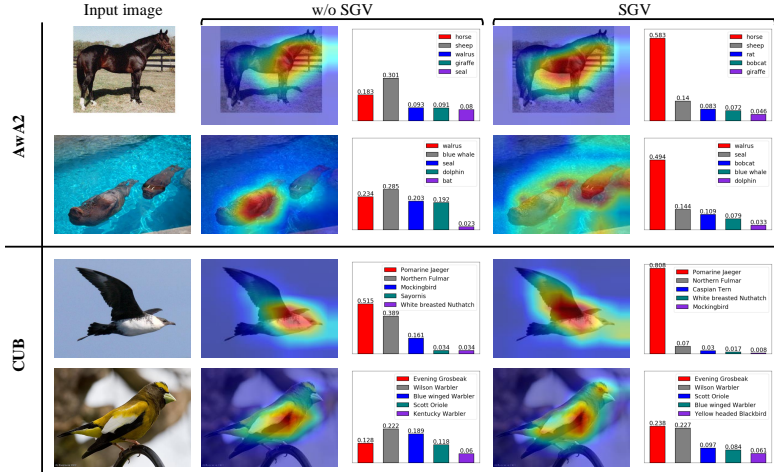
Fig. 8. The Grad-CAM visualization [36] and classification probabilities of SGV^{LT}-101 (w/ SGV) and LFGAA (w/o SGV). The **red** bar denotes the classification probability of target class, and the others indicate top-4 ambiguous classes.

It can be seen that a different degree of ambiguity coincide directly with the classification probabilities, and the attribute word vector output by the SGV rarely appears near independent points of ambiguity. Moreover, the ambiguity class with a high coincidence with the target attribute space can be effectively disambiguated by the SGV, whereas the model without an SGV fails. For the ambiguity class with a lower coincidence, the SGV also significantly reduces the interference of an ambiguous output.

The phenomenon shown in Figure 8 confirms that there are implicit semantic correlations between the visual features in various regions, which is in line with our expectations. The SGV network pays attention to co-occurrence of diverse visual-semantic objects, such as (horses → muscle, meat, fields, etc), (walrus → horns, water, etc), and the head and wing of birds, while the focus area of the model w/o SGV is limited and incomplete. From the perspective of classification probability, the proposed SGV successfully corrects and reduces the ambiguity.

6   Conclusion

In this paper, we propose a semantic graph-enhanced visual network to help visual embedding acquire the ability to model attribute semantic relationships before performing ZSL attribute mapping. It is reasonable to give more semantic support to the visual-semantic bridging and supplement visual embedding with a semantic association representation. To accomplish this, the three complementary designs in the SGV network are applied to the graphed model of the visual features on semantic knowledge graph, performing a semantic regression and

fusing the graph enhanced features to the visual embedding. We perform an in-depth series of experiments across several datasets to validate the performance of our method, and analyse the contribution of each component used. We show how inclusion of an elementary semantic knowledge graph provides a change in attention area that impacts the performance of ZSL.

References

1. Akata, Z., Perronnin, F., Harchaoui, Z., Schmid, C.: Label-embedding for attribute-based classification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 819–826 (2013)
2. Bouma, G.: Normalized (pointwise) mutual information in collocation extraction. In: Proceedings of GSCL. pp. 31–40 (2009)
3. Bronstein, M.M., Bruna, J., LeCun, Y., Szlam, A., Vandergheynst, P.: Geometric deep learning: going beyond euclidean data. IEEE Signal Processing Magazine **34**(4), 18–42 (2017)
4. Chao, W.L., Changpinyo, S., Gong, B., Sha, F.: An empirical study and analysis of generalized zero-shot learning for object recognition in the wild. In: European Conference on Computer Vision. pp. 52–68. Springer (2016)
5. Chen, J.J., Pan, L., Wei, Z., Wang, X., Ngo, C.W., Chua, T.S.: Zero-shot ingredient recognition by multi-relational graph convolutional network. In: Proceedings of the 34th AAAI Conference on Artificial Intelligence (2020)
6. Chen, L., Zhang, H., Xiao, J., Nie, L., Shao, J., Liu, W., Chua, T.S.: Sca-cnn: Spatial and channel-wise attention in convolutional networks for image captioning. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5659–5667 (2017)
7. Chen, R., Chen, T., Hui, X., Wu, H., Li, G., Lin, L.: Knowledge graph transfer network for few-shot recognition. In: Proceedings of the 34th AAAI Conference on Artificial Intelligence (2020)
8. Chen, T., Lin, L., Chen, R., Wu, Y., Luo, X.: Knowledge-embedded representation learning for fine-grained image recognition. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence. pp. 627–634. AAAI Press (2018)
9. Chen, T., Xu, M., Hui, X., Wu, H., Lin, L.: Learning semantic-specific graph representation for multi-label image recognition. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 522–531 (2019)
10. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
11. Farhadi, A., Endres, I., Hoiem, D., Forsyth, D.: Describing objects by their attributes. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. pp. 1778–1785. IEEE (2009)
12. Frome, A., Corrado, G.S., Shlens, J., Bengio, S., Dean, J., Ranzato, M., Mikolov, T.: Devise: A deep visual-semantic embedding model. In: Advances in neural information processing systems. pp. 2121–2129 (2013)
13. Fu, Y., Hospedales, T.M., Xiang, T., Gong, S.: Transductive multi-view zero-shot learning. IEEE transactions on pattern analysis and machine intelligence **37**(11), 2332–2345 (2015)
14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
15. Jiang, H., Wang, R., Shan, S., Chen, X.: Learning class prototypes via structure alignment for zero-shot recognition. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 118–134 (2018)
16. Jiang, H., Wang, R., Shan, S., Chen, X.: Transferable contrastive network for generalized zero-shot learning. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 9765–9774 (2019)

17. Kampffmeyer, M., Chen, Y., Liang, X., Wang, H., Zhang, Y., Xing, E.P.: Rethinking knowledge graph propagation for zero-shot learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 11487–11496 (2019)

18. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)

19. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907 (2016)

20. Kodirov, E., Xiang, T., Gong, S.: Semantic autoencoder for zero-shot learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3174–3183 (2017)

21. Kumar Verma, V., Arora, G., Mishra, A., Rai, P.: Generalized zero-shot learning via synthesized examples. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4281–4289 (2018)

22. Lampert, C.H., Nickisch, H., Harmeling, S.: Learning to detect unseen object classes by between-class attribute transfer. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. pp. 951–958. IEEE (2009)

23. Li, G., Muller, M., Thabet, A., Ghanem, B.: Deepgcns: Can gcns go as deep as cnns? In: Proceedings of the IEEE International Conference on Computer Vision. pp. 9267–9276 (2019)

24. Li, K., Zhang, Y., Li, K., Li, Y., Fu, Y.: Visual semantic reasoning for image-text matching. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4654–4662 (2019)

25. Li, Y., Zhang, J., Zhang, J., Huang, K.: Discriminative learning of latent features for zero-shot recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7463–7471 (2018)

26. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2117–2125 (2017)

27. Liu, Y., Guo, J., Cai, D., He, X.: Attribute attention for semantic disambiguation in zero-shot learning. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 6698–6707 (2019)

28. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems. pp. 3111–3119 (2013)

29. Morgado, P., Vasconcelos, N.: Semantically consistent regularization for zero-shot recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6060–6069 (2017)

30. Ni, J., Zhang, S., Xie, H.: Dual adversarial semantics-consistent network for generalized zero-shot learning. In: Advances in Neural Information Processing Systems. pp. 6143–6154 (2019)

31. Niu, L., Cai, J., Veeraraghavan, A., Zhang, L.: Zero-shot learning via category-specific visual-semantic mapping and label refinement. IEEE Transactions on Image Processing **28**, 965–979 (2019)

32. Patterson, G., Hays, J.: Sun attribute database: Discovering, annotating, and recognizing scene attributes. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition. pp. 2751–2758. IEEE (2012)

33. Romera-Paredes, B., Torr, P.: An embarrassingly simple approach to zero-shot learning. In: International Conference on Machine Learning. pp. 2152–2161 (2015)

34. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International journal of computer vision **115**(3), 211–252 (2015)
35. Schroff, F., Kalenichenko, D., Philbin, J.: Facenet: A unified embedding for face recognition and clustering. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 815–823 (2015)
36. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Gradcam: Visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE international conference on computer vision. pp. 618–626 (2017)
37. Socher, R., Ganjoo, M., Manning, C.D., Ng, A.: Zero-shot learning through cross-modal transfer. In: Advances in neural information processing systems. pp. 935–943 (2013)
38. Song, J., Shen, C., Yang, Y., Liu, Y., Song, M.: Transductive unbiased embedding for zero-shot learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1024–1033 (2018)
39. Sun, C., Shrivastava, A., Vondrick, C., Sukthankar, R., Murphy, K., Schmid, C.: Relational action forecasting. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 273–283 (2019)
40. Tipping, M.E., Bishop, C.M.: Probabilistic principal component analysis. Journal of the Royal Statistical Society: Series B (Statistical Methodology) **61**(3), 611–622 (1999)
41. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The Caltech-UCSD Birds-200-2011 Dataset. Tech. Rep. CNS-TR-2011-001, California Institute of Technology (2011)
42. Wan, Z., Chen, D., Li, Y., Yan, X., Zhang, J., Yu, Y., Liao, J.: Transductive zero-shot learning with visual structure constraint. In: Advances in Neural Information Processing Systems. pp. 9972–9982 (2019)
43. Wang, J., Wang, W., Wang, L., Wang, Z., Feng, D.D., Tan, T.: Learning visual relationship and context-aware attention for image captioning. Pattern Recognition **98**, 107075 (2020)
44. Wang, X., Girshick, R., Gupta, A., He, K.: Non-local neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7794–7803 (2018)
45. Wang, X., Ye, Y., Gupta, A.: Zero-shot recognition via semantic embeddings and knowledge graphs. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6857–6866 (2018)
46. Xian, Y., Lampert, C.H., Schiele, B., Akata, Z.: Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly. IEEE transactions on Pattern Analysis and Machine Intelligence (2018)
47. Xie, G.S., Liu, L., Jin, X., Zhu, F., Zhang, Z., Qin, J., Yao, Y., Shao, L.: Attentive region embedding network for zero-shot learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 9384–9393 (2019)
48. Xu, X., Cao, H., Yang, Y., Yang, E., Deng, C.: Zero-shot metric learning. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence. pp. 3996–4002. AAAI Press (2019)
49. Ye, M., Guo, Y.: Progressive ensemble networks for zero-shot recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 11728–11736 (2019)

50. Zhang, L., Li, X., Arnab, A., Yang, K., Tong, Y., Torr, P.H.: Dual graph convolutional network for semantic segmentation. In: Proceedings of the 30th British Machine Vision Conference (2019)
51. Zhang, Z., Saligrama, V.: Zero-shot learning via semantic similarity embedding. In: Proceedings of the IEEE international conference on computer vision. pp. 4166–4174 (2015)
52. Zhu, Y., Xie, J., Tang, Z., Peng, X., Elgammal, A.: Semantic-guided multi-attention localization for zero-shot learning. In: Advances in Neural Information Processing Systems. pp. 14917–14927 (2019)