Graph-Based Visual-Semantic Entanglement Network for Zero-Shot Image Recognition

Yang Hu , *Student Member, IEEE*, Guihua Wen , Adriane Chapman , Pei Yang, *Member, IEEE*, Mingnan Luo, Yingxue Xu, Dan Dai , and Wendy Hall

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 modeling capabilities to the ZSL task, its visual features have severe pattern inertia and lack of representation of semantic relationships, which leads to severe bias and ambiguity. In response to this, we propose the Graph-based Visual-Semantic Entanglement Network to conduct graph modeling of visual features, which is mapped to semantic attributes by using a knowledge graph, it contains several novel designs: 1. it establishes a multi-path entangled network with the convolutional neural network (CNN) and the graph convolutional network (GCN), which input the visual features from CNN to GCN to model the implicit semantic relations, then GCN feedback the graph modeled information to CNN features; 2. it uses attribute word vectors as the target for the graph semantic modeling of GCN, which forms a self-consistent regression for graph modeling and supervise GCN to learn more personalized attribute relations; 3. it fuses and supplements the hierarchical visual-semantic features refined by graph modeling into visual embedding. Our method outperforms state-of-the-art approaches on multiple representative ZSL datasets: AwA2, CUB, and SUN by promoting the semantic linkage modelling of visual features.**

Index Terms—**Zero-shot learning, visual-semantic modeling, graph convolutional network, semantic knowledge graph, attribute word embedding.**

I. INTRODUCTION

ZERO-shot learning (ZSL) presents a new paradigm for image recognition, whose goal is to train models to
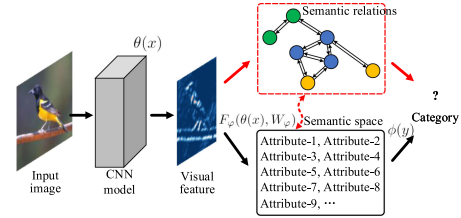
Fig. 1. Illustrating the strategy of graph-based visual-semantic embedding. The red lines and dashed box represent the modelling of implicit semantic links in visual features and help visual-semantic bridging complete the category mapping.

recognize categories that have never been seen before [1]. Because many instances in datasets of the real world do not have associated classes, ZSL is required to make computer vision (CV) more generalized [2]–[6]. In typical supervised learning, some samples are unlabelled and need to be classified. However, in ZSL, the classifying objects are some entire categories (unseen categories), and the only clue that connects the seen and unseen classes is the structured attribute assigned to each class. In recent years, architectures using deep convolutional networks (CNN) are used as the ZSL embedding subnet [5], [7]–[9]. CNN based ZSL models are usually pre-trained by ImageNet [10] to earn the initial visual perception ability [7], [8], [11]. However, such models only provide mappings between classes and observed patterns; knowledge about relationships within the attributes of seen and unseen classes are not utilized.

The components outlined in black in Fig. 1 shows how current ZSL model uses attributes. In existing ZSL methods, there are two types of settings: the inductive setting methods that have no access to unseen classes [11]–[13] and transductive setting methods [7], [14], [15] that have access to unseen samples without labels. However, in both the inductive and transductive settings, visual image modeling and attribute mapping modeling are separate. For instance, 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 distribution vector of the attributes to a class y. $\mathcal{F}(x, y, W_\varphi) = \mathcal{F}_\varphi(\theta(x), W_\varphi)\phi(y)$ is the score function of the classification. The only connection between the visual and the semantic embeddings is the visual-semantic bridging $\mathcal{F}_\varphi(\cdot, W_\varphi)$, which is usually constructed by a few fully-connected layers (FC). Facing visual embedding $\theta(\cdot)$ with strong pattern inertia, $\mathcal{F}_\varphi(\cdot, W_\varphi)$ bears an excessive modelling pressure and hard to reverse prediction bias. Although the latent attributes mechanism reduces
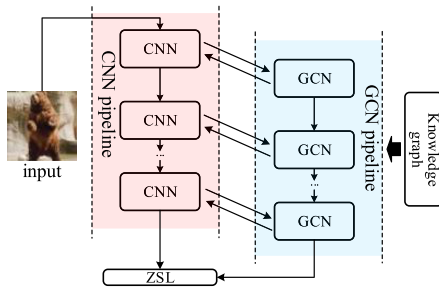
Fig. 2. Illustrating the structure of Graph-based Visual-Semantic Entanglement. The left is the CNN pipeline, which is used to model the visual feature of the input image, and the right is the GCN pipeline, which is used to model the semantic relationship in the visual features and provide feedback. A pre-set knowledge graph supports the GCN pipeline.

the skew caused by the attribute inertia [9], [16], the existing ZSL models still ignore the implicit semantic linkages in visual features.

In order to address the problem of typical visual features which lack knowledge about relationships within the attributes, we must utilize knowledge of the attribute connections to complement the mapping, modeling, and fusion of implicit semantic relationships, as shown in red in Fig. 1. To do this, we propose the **Graph-based Visual-Semantic Entanglement** (GVSE) network. As illustrated in Fig. 2, GVSE network mainly proposes the visual-semantic entanglement structure: There should have a mechanism to map the concept in the attribute space with the semantics representation in the image representation. Attribute-based graph modeling can directly interact with semantic information in visual features. To utilize the relationships amongst attributes found in the semantic knowledge graph, we build the architecture with CNN and graph convolutional network (GCN) [17] two entangled pipelines, we map CNN visual features to attribute-based knowledge graphs, then utilize GCN to model the implicit semantic relations of intermediate visual features and feedback to the CNN. GVSE network also contains some effective designs: 1. To provide better sets of characterized attribute relations for the model, we create a self-consistent system with GCN modelling by creating attribute word vectors to achieve semantic regression. 2. 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 **Graph-based Visual-Semantic Entanglement** network, in which, GCN remodels the CNN visual features with an pre-set knowledge graph contains attribute relational information, and feedbacks semantic relations to the deep visual CNN pipeline.

2) We design to utilize the attribute word vectors as the regression output of the GCN pipeline, which offers a self consistent supervision for semantic graph modelling and helps the model

to mine more personalized attribute relationships based on the initialized attribute knowledge graph.

3) We propose to build the skip-connection from GCN block to visual embedding, which combines the graph remodeled features into visual embedding to enhance the semantic relational representation before we use it in ZSL prediction.

4) We evaluate the proposed GVSE network and supporting techniques. The proposed method achieves state-of-the-art performance on ZSL datasets: AwA2 [18], CUB [19], and SUN [20]. We perform ablation studies and visualization analysis to validate the effectiveness of our designs.

II. RELATED WORKS

A. Zero-Shot Learning

Zero-shot learning (ZSL) uses semantic attributes to connect the search space of unseen objects. The initiatory 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) [1]. The prediction of ZSL relies on a shared semantic space which can be user-defined attribute annotations [21] or unsupervised word embeddings [22].

To achieve targets: 1. removing the gap between attribute predictions to the final category; 2. correcting assumptions that attributes are independent of each other, ESZSL [13] and SCoRe [23] propose to use a multi-layer neural network to train the visual-semantic bridge. SAE [2] introduces external word embeddings as an assistant. However, the "*domain shift*" [24] shows that different categories have distinct visual expressions on some attributes. It causes severe category bias, especially in generalized ZSL setting [25], which mixes the seen and unseen classes in a search space. Methods [5], [26] add regulation in the loss function. AEZSL [27] generates embedding spaces from multiple perspectives, and PREN [8] uses the ensemble method to fuse them. The generative models like adversarial learning [3], [6], [28] and variational auto-encoder [29], [30] are also applied in ZSL to generate more vivid embedding spaces. However, such models usually have inadequate training stability, which has a high parameter adjustment requirement for researchers and is not convenient for application to more fields. The attention-based methods highlight detailed parts. However, they still miss the semantic relations [11], [31]. The transductive ZSL methods [7], [8], [15], [32] even learn a part of unlabeled unseen class samples to remedy the natural lacking of inference clues in visual-semantic modelling. Moreover, LDF [9] suggests that human-defined attributes cannot fully depict semantic information and propose the latent attribute mechanism to learn the pseudo-attributes automatically. LFGAA [16] adds attention values for both human-defined and pseudo attributes. AMS-SFE [33] uses manifold modeling on the expanded semantic feature.

Instead, we argue that another vital reason for the "domain shift" is that the connection between visual features and semantic space is too weak. Specifically, visual features and semantic space are independent of each other, and the semantic connection between visual features is not reflected. The GVSE has obvious

differences with the existing ZSL methods: 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 GVSE network introduces the unbiased knowledge graph to supervise the visual modelling and uses the GCN to learn the cognition of semantic relations.

Some methods also use external semantic ontology [2], [23] or knowledge graph [4], [34]–[36] to assist ZSL models. However, they are still different from GVSE: 1. The graph in all these works is based on the category relations, while we use attributes as nodes to model the attribute relationship. We consider that attribute-based graph modeling can directly map with the latent semantics in visual features. Besides, SEKG [34] and DGP [4] rely on the topology of pre-built ontology WordNet; 2. They use the tandem structure to combine GCN graph modeling and CNN visual modeling modules, which cannot feed the semantic graph information back to the visual features. From another perspective, the intermediate visual features have a rare chance to be modeled on the semantic knowledge graph.

B. 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 [17], [37]. The benefit of visual modelling applications includes: various recognition tasks [38], [39], activity recognition [40], object detection [41] and segmentation [42], as well as visual point cloud data modelling [43]. Researchers use a GNN to model the links between category labels [35], [44], [45] or integrate external knowledge [38], [46]. Graph learning also enhances the representation of the connections between visual objects and semantic entities and improves detection and segmentation [42]. Visual-semantic interaction tasks like visual question & answer (VQA) and image caption are emphasised application areas of a GNN [47]–[49], 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 [17] is the most common method of graph learning. Also, abundant studies can confirm that implicit semantic linkages exist in visual features [49], [50]. Graph modeling is a powerful means of loading structured knowledge into deep models, especially for the ZSL model with a great demand for semantic relationship modeling.

The above methods are significantly different from the proposed GVSE network: 1. In the existing studies, 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; 3. Visual modeling and graph modeling in the existing studies are concatenated in series; the previous part modelling bias will easily affect subsequent modeling. By contrast, the GVSE method uses a GCN to reconstruct and readjust deep visual representations, which uses parallel pipelines to model visual and semantic features separately, with sufficient feedback and robust interactivity. It

TABLE I
EXPLANATION OF THE SPECIAL NOTATIONS UTILIZED IN THIS PAPER

Notation	Meaning
$\mathcal{N}(\cdot)$	The normalization function within $[0, 1]$
$(\cdot)^{\top}$	Matrix transpose
$< \cdot, \cdot >$	A collaborative operating pair consisting of two elements
$[\sim, l]$	For a sequence, all elements from 1 to l
$[l, \sim]$	For a sequence, all elements from l to the end
$[\sim]$	All elements in a sequence
$\bowtie$	The concatenation operation
$[\cdot]_+$	Equivalent to $\max(0, \cdot)$

enhances the ability to model semantic relationships in visual models while retaining the original visual representation.

III. PROPOSED METHOD

A. 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 $\mathcal{Y}^{\mathcal{U}}$ 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: $\forall y_i \exists < Att_1, \dots, Att_m >$ as the only bridge between them and $y_i \in \mathcal{Y}$. Att_i refers to the attribute which is usually a word or concept, 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 $\mathcal{Y}$.

Table I lists some notations and their interpretations that specifically appear in this paper.

B. Overview Framework

In order to achieve the goal of visual-semantic entangled feature modeling for ZSL, the following steps must be taken:

1. **Construction of Semantic Knowledge Graph:** Construct the semantic knowledge graph for the execution of GCN. The knowledge graph extracts co-occurrence relations of attributes.

2. **Establish the Visual-Semantic Dual-pipe Network Structure:** Establish a dual-pipe network structure with clear responsibilities as Fig. 2, in which CNN is responsible for conventional image visual modeling, and GCN is responsible for graph modeling of semantic relations of visual features.

3. **Design the Entanglement Strategy of CNN and GCN:** Establish the interaction functions between CNN visual modeling and GCN semantic modeling. GCN receives visual features from CNN as input, and CNN gets GCN semantic information to further optimize visual features.

4. **Fusion of Graph Semantic Encoding and Visual Representation:** In order to further strengthen the previous feature representation of the ZSL bridging, we merge the semantic graph modeled features from the GCN blocks into the final visual embedding.
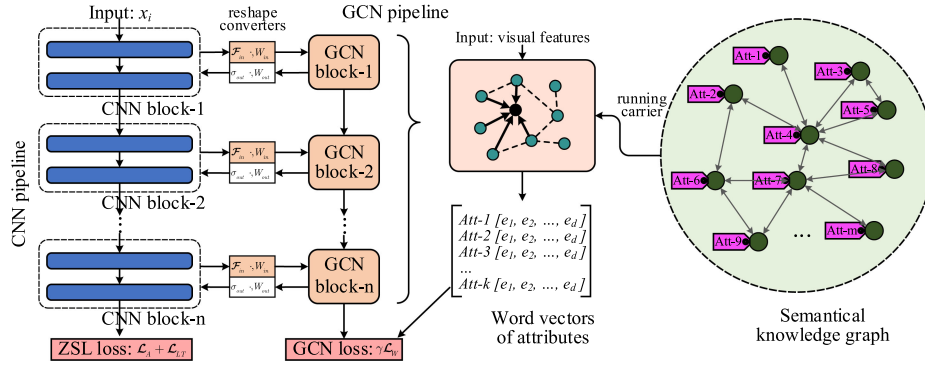
Fig. 3. Overview framework of the GVSE network. The leftmost area is the CNN pipeline, which performs visual modeling for the images. The middle shows the GCN pipeline, which models the implicit semantic relationships in visual features. The reshape converters regular the dimension of visual feature input and semantic information feedback, which are used to support dual-pipe interaction. The rightmost area is the semantic knowledge graph, the vertexes of which refer to the attributes. It is the "running carrier" for GCN pipeline: the graph convolution algorithm is running based on the semantic knowledge graph.

Fig. 3 shows the overview framework of the GVSE network, which illustrates the semantic knowledge graph for attributes, CNN visual modeling pipeline, GCN semantic modeling pipeline, the target outputs of dual-pipe, and the interaction support modules for dual-pipe entanglement.

1) Semantic Knowledge Graph: Since GCN needs to run on a pre-defined graph structure, before constructing the dual-pipe visual-semantic neural network architecture, we first introduce the construction strategy of the semantic knowledge graph. This knowledge graph will have the following characteristics: 1. The visual features are directly mapped with attributes. Thus, the knowledge graph is expected to represent the fundamental relationship of attributes; 2. The establishment of the knowledge graph is off-line and convenient; 3. The establishment of the knowledge graph can apply to the various scenes, without being restricted by the ontology adapted to the data set.

Co-occurrence relationship is one of the basic linkages between attributes. It can be obtained by counting how often the attribute appears together in each category. Instead of applying WordNet as a prior knowledge graph [4], [34], we construct the semantic knowledge graph based on attribute co-occurrence relations concerning all categories. The attributes with co-occurrence relations will have a high probability of existing in the same category's visual representation.

A 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 construction of our knowledge graph requires the attribute space indicates the unequivocal concepts. In our semantic knowledge graph, the attributes are used to define the vertices, and the point-wise mutual information (PMI) [51] is used to calculate the attributes co-occurrence and determine the connection between attribute vertices, as follows:

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

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 the co-occurrence probability $p(v_i, v_j)$ is the ratio of the category

Fig. 4. Illustration of attribute linkage 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.

amount with both v_i and v_j to the total number of categories. $\mathcal{N}$ denotes the normalization function within $[0, 1]$. Fig. 3 illustrates the construction strategy of the semantic knowledge graph, and the edge is built between the vertices with a PMI of higher than the threshold δ, as:

$$l_{i,j} = \begin{cases} 1, & \mathcal{PMI}(v_i, v_j) > \delta \\ 0, & \text{else} \end{cases}, \qquad (2)$$

Fig. 4 shows an example of edge construction in our semantic knowledge graph. After obtaining the pre-designed semantic knowledge graph $\mathcal{G}_{att} = (\mathcal{V}, \mathcal{E})$, the dual-pipe network structure is introduced in the followed section.

2) Dual-Pipe Network Structure: The two modeling pipelines we designed to have a clear division of labour. We first define the CNN pipeline for visual feature modeling as:

$$\theta(x_i) = \mathcal{F}_{conv}(x_i, W_\theta), \qquad (3)$$

where x_i refers to the input and $\theta(x_i)$ is the visual embedding modeled and provided by CNN pipeline $\mathcal{F}_{conv}$, which is parameterized by weights W_θ. In our study, the CNN pipeline $\mathcal{F}_{conv}$ can be constructed with any of existing popular CNN backbone [52], [53].

The target prediction output of the CNN pipeline is the ZSL classification score $p(y|x_i)$, and it is calculated as follows:

$$p(y|x_i) = \mathcal{F}_\varphi(\theta(x_i), W_\varphi)^\top \phi(y), \qquad (4)$$
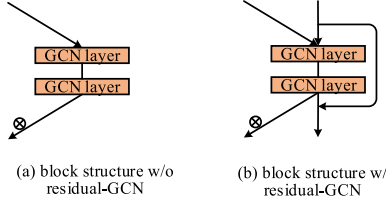
Fig. 5. Comparison of different GCN blocks. (a) The GCN blocks without residual structure; (b) The GCN blocks with residual skip-connection.

where $\mathcal{F}_{\varphi} : \mathbb{R}^{d_v} \to \mathbb{R}^m$ is the attribute prediction layer with parameter $W_{\varphi} \in \mathbb{R}^{d_v \times m}$, and $\phi(y) \in \mathbb{R}^{m \times |\mathcal{Y}|}$ denotes the attribute distribution. d_v is the dimension of visual features, m and $|\mathcal{Y}|$ refer to the numbers of attributes and categories.

Based on the premise that there are implicit semantic linkages in visual features, and this has been found in other CV studies [54], we design the GCN pipeline to model the semantic relations in visual features. As illustrated by Fig. 3, note that we have feature maps $X^{(l)} = \mathcal{F}_{conv}^{[\sim,l]}(x_i, W_{\theta}^{[\sim,l]})$ from the l-th block of CNN pipeline, where $[\sim, l]$ means "from 1 to l". We use $X^{(l)}$ as the input of the corresponding GCN block $\mathcal{F}_{\mathcal{G}}^{(l)}$, as follows:

$$f_{\mathcal{G}}^{(l)} = \mathcal{F}_{\mathcal{G}}^{(l)} \left(X^{(l)}, < \mathcal{G}_{att}, W_{\mathcal{G}}^{(l)} > \right), \qquad (5)$$

where $f_{\mathcal{G}}^{(l)}$ is the output of the l-th GCN block, $W_{\mathcal{G}}^{(l)}$ is the parameters of $\mathcal{F}_{\mathcal{G}}^{(l)}$ that running on graph $\mathcal{G}_{att}$. The elements in $< \cdot, \cdot >$ must be utilized together. Each GCN block contains 2 GCN layers, which is defined in [17], as follows:

$$\mathcal{H}^{(i+1)} = \sigma \left(D^{-1} G_a \mathcal{H}^{(i)} W_{\mathcal{G}}^{(i)} \right) \qquad (6)$$

where $\mathcal{H}^{(i)}$ and $\mathcal{H}^{(i+1)}$ denotes any two layers in GCN, D and G_a are the degree and adjacency matrices of the pre-set knowledge graph $\mathcal{G}_{att}$, $W_{\mathcal{G}}^{(i)}$ denotes the parameters of i-th GCN layers, and σ is an activation function. In the GCN pipeline, we assign clear attribute definitions to the initially disordered visual features $X^{(l)}$, making GCN: $\mathcal{F}_{\mathcal{G}}^{(l)}$ models them according to the relations described by the graph $\mathcal{G}_{att}$, thereby activating the implicit semantic linkage in $X^{(l)}$.

The GCN pipeline connects all GCN blocks in series, to solve the gradient diffusion on the GCN pipeline, we adopt the residual GCN structure [43] shown in Fig. 5(b), as follows:

$$f_{\mathcal{G}}^{(l+1)} = F_{\mathcal{G}}^{(l+1)} \left(X^{(l+1)}, f_{\mathcal{G}}^{(l)}, < \mathcal{G}_{att}, W_{\mathcal{G}}^{(l+1)} > \right) + f_{\mathcal{G}}^{(l)}. \qquad (7)$$

Since we record the GCN pipeline as $\mathcal{F}_{\mathcal{G}}$, its target output is designed to the attribute word vectors of category:

$$\mathcal{A}(x_i) = \mathcal{F}_{\mathcal{G}} (x_i, < \mathcal{G}_{att}, W_{\mathcal{G}} >), \qquad (8)$$

where $W_{\mathcal{G}}$ is the parameters of the GCN pipeline, and

$$\mathcal{A}(x_i) = [a_1(x_i), a_2(x_i), \ldots, a_k(x_i)], \qquad (9)$$

which is the set of attribute word vectors. $a_1(x_i) \ldots a_k(x_i)$ are the word vectors of attributes $Att_1 \ldots Att_k$, these attributes belong to the ground-truth category y_i.

How we get the attribute word vector: in ZSL data sets, there are several attributes belongs to each category so that

we can consider each category $y \in \mathcal{Y}$ as a bag of attribute $\{Att_1, Att_2, \ldots, Att_k\}$, where k is the number of attributes that the category y has, then we can consider all categories $\mathcal{Y}$ as corpora. With this corpora, we can conveniently train a word embedding $\mathcal{M}_e$ with existing language model tools [22], [55]. Like the knowledge graph $\mathcal{G}_{att}$, the acquisition of the word embedding $\mathcal{M}_e$ is off-line and fast. We can query the word vector $a_j = [e_j^1, e_j^2, \ldots, e_j^d] \in \mathbb{R}^d$ of an arbitrary attribute Att_j from $\mathcal{M}_e$, where d is the fixed dimension of word embedding.

The reasons why we set the target output of the GCN pipeline as above are: 1. As a model based on vocabulary co-occurrence, the word vectors can form self-consistent regression with above semantic knowledge graph $\mathcal{G}_{att}$; 2. The semantic knowledge graph $\mathcal{G}_{att}$ only provides initial attribute relational information based on the average strong attribute co-occurrence. The GCN pipeline needs to learn more personalized attribute relations with word embedding supervision.

3) Entanglement Strategy of CNN and GCN: Only constructed the CNN pipelines $\mathcal{F}_{conv}$ and GCN pipeline $\mathcal{F}_{\mathcal{G}}$ is not enough. We design the entanglement strategy between them based on the following motivations: 1. The modeling process of CNN and GCN pipelines needs interaction and synchronization; 2. Hierarchical visual features need to receive information on semantic relational modeling. This section introduces the dual-pipe entanglement strategy in detail from two directions: CNN to GCN and GCN to CNN.

For **CNN to GCN**, we have described that using feature maps $X^{(l)}$ from CNN block as the input of GCN block. However, the shape of $X^{(l)}$ may not be suitable to the GCN input, so we need to reshape $X^{(l)}$ first, as:

$$X^{(l)} = \mathcal{F}_{in}^{(l)} \left(X^{(l)}, W_{in}^{(l)} \right), \qquad (10)$$

where $\mathcal{F}_{in}^{(l)}$ is the transform function with weights $W_{in}^{(l)}$.

In addition, starting from the second GCN block $\mathcal{F}_{\mathcal{G}}^{(2)}$, the concatenation operation $\bowtie$ is used to merge the feature maps $X^{(l)}$ and the output $f_{\mathcal{G}}^{l-1}$ of the previous GCN block as the input. Thus we can update Equations 5 and 7 to:

$$f_{\mathcal{G}}^{(l)} = \begin{cases} \mathcal{F}_{\mathcal{G}}^{(l)} \left(X^{(l)}, < \mathcal{G}_{att}, W_{\mathcal{G}}^{(l)} > \right), & l = 1 \\ \mathcal{F}_{\mathcal{G}}^{(l)} \left(X^{(l)} \bowtie \mathcal{F}_{sq} \left(f_{\mathcal{G}}^{(l-1)} \right), & l \geq 2 , \\ \quad < \mathcal{G}_{att}, W_{\mathcal{G}}^{(l)} > \right) + f_{\mathcal{G}}^{(l-1)}, \end{cases} \qquad (11)$$

where $\mathcal{F}_{sq}$ is a squeeze function, which reduces the dimension of $f_{\mathcal{G}}^{(l-1)}$ to d for convenient, it saves the calculation of $\mathcal{F}_{\mathcal{G}}^{(l)}$.

For **GCN to CNN**, we apply the gate mechanism to feedback the semantic relationship modeling information of the GCN pipeline to the visual features of the CNN pipeline, as follows:

$$\tilde{X}^{(l)} = \sigma_{out} \left(f_{\mathcal{G}}^{(l)}, W_{out}^{(l)} \right) \otimes X^{(l)}, \qquad (12)$$

where, the weight $W_{out}^{(l)}$ is used to align the dimension of the graph modeling information $f_{\mathcal{G}}^{(l)}$ with the feature graph $X^{(l)}$, and its role is similar to $W_{in}^{(l)}$. σ_{out} is an activation function and outcomes value within range $[0, 1]$, operation $\otimes$ is the element-wise multiplication, and $\tilde{X}^{(l)}$ is the new visual feature maps,

which will continue feed to the subsequent CNN pipeline, as: $\mathcal{F}_{conv}^{[l+1,\sim]}(\tilde{X}^{(l)}, W_\theta^{[l+1,\sim]})$, where $[l+1,\sim]$ meas "from $l+1$ to the end".

The entanglement strategy of the proposed GVSE network is shown in Fig. 3, and it makes the dual-pipe structure of the GVSE network get full interaction in addition to their respective precise modeling functions. Our strategy differs from the other ZSL models based on attention methods [3], [16], [31], [56] in particular: 1. Utilizing graph modeling to optimize visual implicit semantic information; 2. CNN features of various layers will get optimization and feedback.

The motivation for GCN semantic graph modeling on each CNN layer is that we consider the visual features of various CNN block contain implicit semantic relationship information, and modeling the semantic relationship of each CNN block supports to integrate the visual-semantic modeling into the entire CNN pipeline, which provides the semantic-rich ZSL visual embedding with complete optimization. In other words, our semantic graph modeling of visual features will not be squeezed into one block. It will be evenly distributed in the hierarchy, to share semantic modelling pressure and obtain hierarchical visual-semantic information.

4) Integrate Hierarchical Semantic Features Into Visual Embedding: To further enhance the final visual embedding, we perform one last step. As illustrated in Fig. 5, inspired by the pyramid structure [57], we merge the semantic features from GCN pipeline into the final visual embedding $\theta(x)$, i.e. **Semantic Relational Features (SRF)**. Its advantages are: 1. Help the loss back-propagation to the GCN blocks; 2. Provide more semantic information for the latent attributes [9], [16].

Note that the final visual embedding is concentrated by global average pooling as:

$$\theta(x_i) = \frac{1}{R \times C} \sum_{r=1}^{R} \sum_{c=1}^{C} \mathcal{F}_{conv}^{[\sim]}(x_i, W_\theta)[r, c], \quad (13)$$

where $\mathcal{F}_{conv}^{[\sim]}$ is the part of CNN pipeline before global pooling, $(R \times C)$ is the shape of final feature maps of CNN pipeline, and $\mathcal{F}_{conv}^{[\sim]}(x_i, W_\theta)[r, c]$ indicates the signal of r-th row and c-th column on CNN feature maps.

The integrated visual embedding $\theta(x)^+$ is obtained as follows:

$$\theta(x)^+ = \theta(x) \bowtie \hat{f}_\mathcal{G}^{(1)} \bowtie \hat{f}_\mathcal{G}^{(2)} \bowtie \cdots \bowtie \hat{f}_\mathcal{G}^{(L)}, \quad (14)$$

where

$$\hat{f}_\mathcal{G}^{(i)} = \frac{1}{k} \sum_{v=1}^{k} \left(\mathcal{F}_{sq}(f_\mathcal{G}^{(i)}[v]) \right), \quad (15)$$

which is utilized to pool the GCN outputs of attribute word vector, and k is the attribute number. Still, $\mathcal{F}_{sq}$ is a squeeze function to reduce the width of GCN outputs.

IV. OPTIMIZATION DETAILS AND ZSL PREDICTION

The GVSE network in this study provides ZSL with a more robust feature of semantic relation representation, which is widely suitable for various ZSL frameworks. Since we have introduced the dual-pipe feedforward feature modeling method of the GVSE network in Section III, in this section, we briefly describe the ZSL prediction method with the GVSE network and the optimization method for GVSE features.

We introduce the execution of the ZSL prediction and optimization from inductive and transductive these two ZSL settings. The proposed GVSE network uses the back-propagation (BP) algorithm [58] for optimization, and the loss function indicate the detailed optimization strategy.

A. Inductive ZSL Setting

Since Eq. 4 formulates the score function of ZSL prediction, we pick the label with maximum score as:

$$y_i^* = \arg\max_{y \in \mathcal{Y}} p(y|x_i) = \arg\max_{y \in \mathcal{Y}} \left(\varphi(x_i)^\top \phi(y) \right), \quad (16)$$

where $\varphi(x_i) = \mathcal{F}_\varphi(\theta(x_i)^+, W_\varphi)$ refers to the visual-semantic projection which is target to attributes.

We also introduce latent attributes (LA) [9], [16] to supplement the imperfect artificially defined attribute space. This mechanism needs to calculate the prototypes of all classes for latent attributes, for seen classes: $\phi_{lat}(y^s) = \frac{1}{N}\sum_i \varphi_{lat}(x_i)$, where $\varphi_{lat}(x_i)$ denotes the discriminated features specifically for the latent attributes, which is elaborated by [9]. For unseen classes, the prototypes can be obtained from a ridge regression:

$$\beta_y^u = \arg\min_{y \in \mathcal{Y}^S} \left\| \phi(y^u) - \sum \beta_y^u \phi(y) \right\|_2^2 + \left\| \beta_y^u \right\|_2^2, \quad (17)$$

$$\phi_{lat}(y^u) = \sum_{y \in \mathcal{Y}^S} \beta_y^u \phi_{lat}(y), \quad (18)$$

then, the ZSL prediction is preformed as:

$$y_i^* = \arg\max_{y \in \mathcal{Y}} \left(\varphi(x_i)^\top \phi(y) + \varphi_{lat}(x_i)^\top \phi_{lat}(y) \right), \quad (19)$$

Since we have the above ZSL prediction process, the loss function is set as follows, we utilize the softmax cross-entropy loss to optimize the visual-semantic bridging, as follows:

$$\mathcal{L}_A = -\frac{1}{N} \sum_i^N \log \frac{\exp\left(\varphi(x_i)^\top \phi(y_i)\right)}{\sum_{y \in \mathcal{Y}^S} \exp\left(\varphi(x_i)^\top \phi(y)\right)}. \quad (20)$$

For discriminated features φ_{lat}, we follow the same strategy in [9], [16] that uses triplet loss [62] to enlarge the inter-class distance and reducing the intra-class distance:

$$\mathcal{L}_{LT} = \frac{1}{N} \sum_i^N \left[\left\| \varphi_{lat}(x_i) - \varphi_{lat}(x_j) \right\|^2 \right.$$
$$\left. - \left\| \varphi_{lat}(x_i) - \varphi_{lat}(x_r) \right\|^2 + \alpha \right]_+, \quad (21)$$

where x_i, x_j, and x_r represent the anchor, positive, and negative sample in a triplet respectively, $[\cdot]_+$ is equivalent to $\max(0, \cdot)$, α is the margin of triplet loss and is 1.0 in all experiments.

As formulated by Eq. 8, since we use attribute word vectors as the target output of GCN pipeline $\mathcal{F}_\mathcal{G}$, the mean-square error is utilized as the loss of this branch, as:

$$\mathcal{L}_W = \frac{1}{N} \sum_i^N \sum_v^{k_i} \left\| \mathcal{F}_{sq}\left(f_\mathcal{G}^{(L)}(x_i) \right)[v] - a_v(y_i) \right\|^2, \quad (22)$$

where $f_{\mathcal{G}}^{(L)}$ indicates the outcome feature from the last GCN block and k_i is the number of attributes belongs to class y_i.

The complete loss function is as follows:

$$\mathcal{L} = \mathcal{L}_A + \mathcal{L}_{LT} + \gamma \mathcal{L}_W. \tag{23}$$

B. Transductive ZSL Setting

We also applied the relation enhanced features learned by GVSE network to three transductive ZSL frameworks: 1. Quasi-Fully Supervised Learning (QFSL) [7]; 2. Self-Adaptation (SA) [16]; 3. Wasserstein-Distance-based Visual Structure Constraint(WDVSc) [15]. These transduced ZSL methods all open access to unseen data to the model, thereby alleviating the "domain shift" phenomenon.

Among them, the SA method needs to be performed in conjunction with LA mechanism [9]; it absorbs the information of unseen category and conducts several offline iterations to update the prototype $\phi_{lat}(y^u)$ and the attribute distribution $\phi(y^u)$ for unseen classes.

The other two transductive ZSL methods are not utilized together with LA mechanism. In QFSL, a bias term $\mathcal{L}_B = \frac{1}{N^u} \sum_{i=1}^{N^u} (- \ln \sum_{y^u \in \mathcal{Y}^u} p(y^u|x_i))$ is added to the loss function to encourage the model to output the classification results of the unseen categories.

In WDVSc, the Wasserstein-distance [63] is utilized to correct the position of the cluster centres of the unseen categories in the semantic transductive space. And it determines the classification result by calculating the distance between the feature distribution and the category cluster centres.

The special design of prediction and optimization in these transductive ZSL methods will not affect the modeling of the GVSE network for the relation-enhanced visual embedding $\theta(x)^+$. Their detailed implementations are described in the original studies [7], [15], [16].

V. EXPERIMENTS

A. Experimental Setup

1) Datasets: We conduct the experiments on three representative ZSL datasets: Animals with Attribute 2 [18] (AwA2), Caltech-UCSD Birds 200-2011 [19] (CUB) and Scene Classification Database [20] (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 [18].

2) 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 [18]. In the CZSL setting, all test samples come from the unseen classes ($y \in \mathcal{Y}^\mathcal{U}$). In the GZSL setting, the test samples come from both seen and unseen classes ($y \in \mathcal{Y}^\mathcal{S} \cup \mathcal{Y}^\mathcal{U}$), and we report the accuracy $\mathcal{ACC}^\mathcal{S}$,

$\mathcal{ACC}^\mathcal{U}$ of seen and unseen test samples and their harmonic mean $\mathcal{H} = \frac{2 \times \mathcal{ACC}^\mathcal{S} \times \mathcal{ACC}^\mathcal{U}}{\mathcal{ACC}^\mathcal{S} + \mathcal{ACC}^\mathcal{U}}$.

3) Implementations: The backbone network ResNet [53] is applied to initialise the Embedding Subnet using the pre-trained weights on ImageNet-1k dataset [64]. Moreover, we test ResNet [53] with 18, 50, and 101 layers as a backbone to demonstrate that the proposed GVSE architecture also works well on the smaller Embedding Subnet. As far as we know, this is the first study to conduct a comparison on ZSL with different backbone scale.

For other hyper-parameters in the GVSE network, the In and Out transform functions $\mathcal{F}_{in}(\cdot)$ and $\sigma_{out}(\cdot)$ in Eqs. 10 and 12 are implemented with a FC mapping, and the activation function of $\sigma_{out}(\cdot)$ is Sigmoid. Each block in GCN pipeline contains two GCN layers with the dimension of 64 and $R_i \times C_i$, where ($R_i \times C_i$) is the shape of the visual feature maps. The dimension d of attribute word vectors is 10. The threshold δ for edge building with measure of PMI, and balancing factor γ are set to 0.75 and 0.5, respectively. We apply the Adam optimizer [65] throughout all experiments. Unless otherwise specified, our GVSE model utilizes the attribute word vector GCN sub-output and STFs by default.

B. Conventional Comparison

1) Overall Performance: CZSL mainly tests the recognition ability of the model for unseen samples. We compare various state-of-the-art transductive ZSL methods [7], [8], [14]–[16], [29], [30], [32] and inductive ZSL approaches [1], [3], [5], [11]–[13], [31], [36], [59]–[61]. In addition to an ordinary GVSE network, we also add the models of GVSE^{LT} (with the latent attributes in [9]), GVSE + T (using the transductive method in [7]), GVSE^{LT} + SA (using the transductive method in [16]) into the competition, and GVSE + WDVSc (using the transductive strategy with cluster center correcting in [15]).

Table II shows that the proposed GVSE network outperforms the existing ZSL methods with both inductive and transductive strategies on CZSL. In most cases, our GVSE network can achieve the best results. Even with fewer layers, the results of GVSE-50, even GVSE-18, remain extremely competitive.

In addition to the overall excellent performance, the graph-enhanced features extracted by the GVSE network can bring the considerable absolute improvements to the baselines, such as GVSE^{LT}-101 improves $2.2\% \sim 5.9\%$ over LFGAA, and GVSE-101 + WDVSc improves $1.8\% \sim 10.0\%$ over WDVSc. These improvements are varying, indicating that the baseline methods only perform well on individual data sets. In contrast, the GVSE model can perform well on all data sets, and the improvement is more obvious for the baseline method's poor condition. Further, The compared methods frequently only show an excellent performance on one dataset, while the GVSE can achieve the best on almost all datasets. We also found that GVSE 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

TABLE II
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 GVSE-18, GVSE-50, AND GVSE-101, THE VISUAL EMBEDDING SUBNET IS RESNET WITH 18, 50, AND 101 LAYERS. BOTH STANDARD SPLIT (SS) AND PROPOSED SPLIT (PS) ARE CONSIDERED [18]. 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 [1]	58.7	46.1	37.5	40.0	38.9	39.9
	SSE [12]	67.5	61.0	43.7	43.9	25.4	54.5
	DEVISE [59]	68.6	59.7	53.2	52.0	57.5	56.5
	ESZSL [13]	75.6	55.1	43.7	53.9	57.3	54.5
	ALE [60]	80.3	62.5	53.2	54.9	59.1	58.1
	CDL [61]	79.5	67.9	54.5	54.5	61.3	63.6
	AREN [3]	**86.7**	67.9	**70.7**	**71.8**	61.7	60.6
	LFGAA [16]	84.3	68.1	67.6	67.6	**62.0**	61.5
	TCN [5]	70.3	71.2	–	59.5	–	61.5
	SGMA [11]	83.5	68.8	70.5	71.0	–	–
	APNet [36]	–	68.0	–	57.7	–	62.3
	DAZLE [31]	–	–	67.8	65.9	–	–
	GVSE-18 (ours)	77.3 ± 0.8	67.5 ± 1.1	68.1 ± 1.2	67.2 ± 0.6	59.2 ± 1.5	59.0 ± 1.1
	GVSE-50 (ours)	82.6 ± 1.6	**71.4 ± 1.2**	69.2 ± 0.9	70.3 ± 1.3	61.6 ± 1.3	61.7 ± 1.0
	GVSE-101 (ours)	**85.8 ± 0.8**	**73.2 ± 1.0**	70.6 ± 1.1	**72.2 ± 1.4**	**64.3 ± 1.0**	**63.8 ± 1.3**
	GVSE^{LT}-101 (ours)	**88.1 ± 0.7**	**74.0 ± 1.5**	**71.4 ± 1.2**	**72.8 ± 1.1**	**64.2 ± 1.6**	**64.8 ± 0.9**
$\mathcal{T}$	SE-ZSL [14]	80.8	69.2	60.3	59.6	64.5	63.4
	QFSL [7]	84.8	79.7	69.7	72.1	61.7	58.3
	PREN [8]	95.7	74.1	66.9	66.4	63.3	62.9
	f-VAEGAN [29]	–	**89.3**	–	**82.6**	–	**72.6**
	LFGAA + SA [16]	94.4	84.8	**79.7**	78.9	64.0	66.2
	WDVSc [15]	**96.7**	87.3	74.2	73.4	67.8	63.4
	HPL [32]	**97.8**	**91.2**	75.3	75.2	**80.4**	**70.4**
	Zero-VAE-GAN [30]	93.8	85.4	69.1	68.9	**68.4**	66.8
	GVSE-50 + T (ours)	91.0 ± 0.7	80.5 ± 0.9	74.2 ± 1.8	76.0 ± 0.9	60.5 ± 0.8	62.6 ± 1.2
	GVSE-101 + T (ours)	93.2 ± 0.9	83.7 ± 0.9	77.6 ± 1.2	79.4 ± 1.4	64.0 ± 1.5	63.8 ± 1.2
	GVSE^{LT}-101 + SA (ours)	96.3 ± 0.8	86.8 ± 0.9	**81.0 ± 1.2**	**82.9 ± 0.9**	65.0 ± 1.2	67.9 ± 1.1
	GVSE-101 + WDVSc (ours)	**98.5 ± 1.1**	**92.3 ± 1.0**	76.9 ± 1.7	76.1 ± 0.9	75.5 ± 1.3	**73.4 ± 0.8**

the GVSE has strong cross-modal semantic modeling capabilities on fine-grained visual features and clearly described attributes.

The GVSE network does not perform as well as HPL on the standard split SUN dataset, and it has no significant advantage over f-VAEGAN on the CUB dataset. The reason is that when the transduction setting opens access to unseen samples, HPL and f-VAEGAN can provide more discriminable prototype representation space and more realistic transductive features. Fortunately, the proposed GVSE has good adaptability and does not conflict with these approaches. We plan to verify the combination of GVSE features and more methods in the future.

2) Comparison of the GVSE vs Other Attention Strategies: The results of ablation study on semantic relationship modeling tools are provided. This experiment is expected to demonstrate the advantages of GCN pipelines for modeling semantic relationships within visual features. The other two options used to compare with the GCN pipeline are: 1. Remove the GCN pipeline, merge the visual features of different depths in the CNN pipeline and directly use the fully connected encoder to model them (Direct), which is illustrated as Fig. 7; 2. The graph convolutional layers in GCN pipeline are replaced with FC layers of the same dimensions (FC encoder). The word vector auxiliary output is available here.

Tables III lists that the GCN pipeline has advantages for the other two FC encoders on all the data sets participating in the experiment, which confirms the positive effect of explicit graph modeling for visual-semantic linkages on category transferring. In **Appendix**, more complete results are provided.
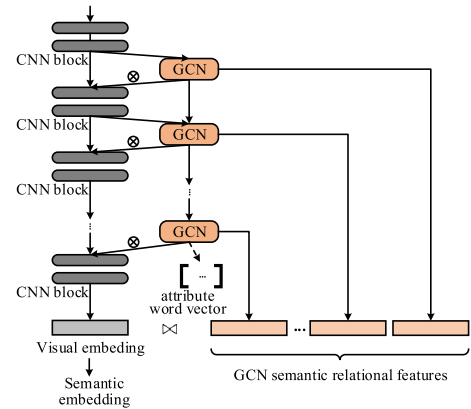
Fig. 6. Illustration of the GVSE network with the semantic relational features (SRFs) that fused into the final visual embedding.
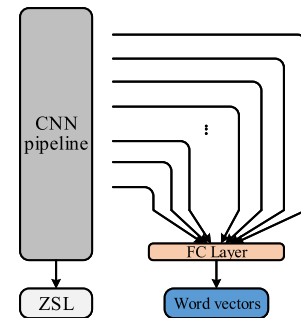
Fig. 7. Illustration of the "Direct" structure in ablation study on semantic modeling tools, which removes the GCN pipeline and merges the visual features of various blocks in the CNN pipeline. It utilizes the fully connected encoder to model them. The word vector auxiliary task gives the direct penalty to the outputs of each CNN block.

Fig. 8. Comparisons on AwA2, CUB, and SUN datasets with the proposed split (PS) (%). We apply the inductive setting and set the latent attribute mechanism [9] and attribute word vector GCN sub-output as the ablation options.

TABLE III
ABLATION RESULTS (%) WITH DIFFERENT SEMANTIC MODELING TOOLS: 1.
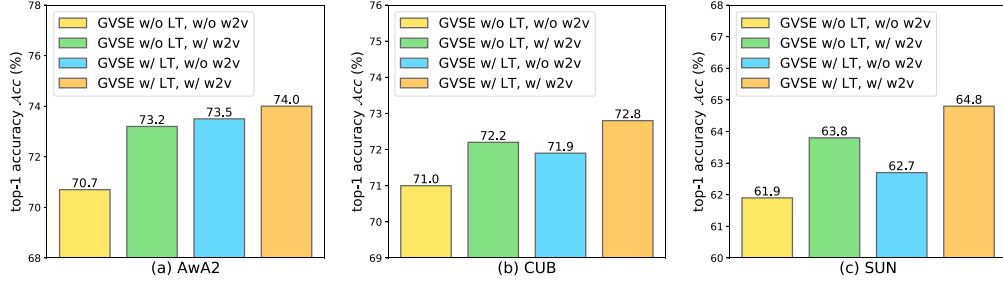ONE FC LAYER DIRECT ENCODER; 2. FC ENCODER ON EACH BLOCK; 3. GCN
PIPELINE (OURS). RESNET-101 IS UTILIZED AS BACKBONE. ACCURACY ($\mathcal{ACC}$)
IN CZSL WITH PROPOSED SPLIT (PS) IS RECORDED

Model	Semantic modeling tool	AwA2	CUB	SUN
	Direct	68.9	68.0	59.7
ResNet-101	FC encoder	69.1	68.4	61.3
	GCN (ours)	**73.2**	**72.2**	**63.8**

TABLE IV
ABLATION RESULTS WITH OR WITHOUT GVSE ON PROPOSED SPLIT (PS) (%).
FOR GVSE$^{(\text{LAST BLOCK})}$, GVSE$^{(\text{EACH STAGE})}$, AND GVSE$^{(\text{EACH BLOCK})}$, THE GCN
BLOCK (IN GCN PIPELINE) IS ADDED FOR EACH CNN BLOCK, EACH STAGE,
AND THE LAST CNN BLOCK

Option	Network	AwA2	CUB	SUN
	B-Net$^{w/o\ GVSE}$	62.7	59.0	56.5
w/o GVSE	Att-Net [66]	63.9	64.2	57.7
	SCA-Net [66]	63.3	65.5	59.0
	CBAM-Net [67]	62.8	65.8	58.1
	GVSE$^{(\text{last block})}$	72.8	70.4	61.9
w/ GVSE	GVSE$^{(\text{each stage})}$	**73.2**	71.8	62.9
	GVSE$^{(\text{each block})}$	72.0	**72.2**	**63.8**

Then, the comparison results of the proposed method and other attention models are provided. As Table IV shows, we compare the GVSE network with other attention networks to validate our graph-modeling effectiveness on ZSL. We chose the inductive version of QFSL [7] (denoted as B-Net$^{w/o\ GVSE}$) as the baseline. To better show the benefit of GVSE, we introduce three baselines Att-Net in [66] with spatial attention, SCA-Net [66] and CBAM-Net [67] and both spatial and channel-wise attention. We also analyse the performance when adding the GCN block to each CNN block, each stage (when the size of the receptive field changes), or the last CNN block (in the **Appendix**, we present the detailed structure of different GCN block adding strategies). Although the typical attention mechanisms can improve the baseline slightly, it is still not able to compare the ability of graph modeling for a semantic representation, and the GVSE shows overwhelming advantages. Moreover, the granularity of the dataset decides requirements for GCN blocks in GVSE's GCN pipeline. On AwA2, fewer GCN blocks achieve the best results, the reason is that semantic attributes are relatively obscure in coarse-grained visual features.

3) Ablation Studies of the Attribute Word Vector Sub-Output:
We validate the improvement from applying the attribute word vector GCN sub-output. As illustrated in Fig. 8, we can see that the regression of the semantic graph modelling by the attribute word vector target output, and this design can significantly improve the performance of visual-semantic modeling. Particularly on fine-grained image recognition datasets, like CUB and SUN, the benefits of word vector 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 GCN sub-output and latent attributes are utilized simultaneously, the classification accuracy reaches its peak, which means that the regression of semantic graph modeling to attribute word vectors supporting ZSL model generates more accurate latent attributes.

4) Ablation Studies on Using of SRF: We conduct the ablation study to validate the effect of SRF and set the mechanism of the latent attribute as another ablation option to test the collaborative performance of SRF and the latent attribute mechanisms [9], [16]. The different structures of various SRF modes are introduced in **Appendix**. Fig. 9 shows that, as SRF concatenates more semantic features into a visual embedding, the performance of the model is continuously improved, and better results are obtained when we utilize latent attributes and STF concurrently. We can conclude that SRF not only complements the visual embedding, it also boosts the modelling of the latent attributes.

In **Appendix**, we provide more comprehensive ablation research results, including all permutations and combinations of SRF feature fusion and different GCN block numbers, and some tests on residual structures' benefits for GCN pipelines.

5) Ablation Studies on the Type of Semantic Knowledge Graph: The ablation studies on knowledge graph types and graph modeling strategies are provided in this section. For knowledge graph, there are three types of graph are introduced. In addition to the Attribute-based graph based on the attribute co-occurrence applied in the proposed method, we also apply the Category-based graph based on the cosine similarity between categories, and the instance k-NN graph established in reference [68]. The cosine similarity $\mathcal{S}im$ between categories is calculated as Eq. 24. For the semantic graph modeling, in addition to the proposed dual-pipeline architecture, the structure of comparison is to place semantic graph modeling in series with the auxiliary output of word vector after visual modeling.

$$\mathcal{S}im(y_i, y_j) = \frac{\phi(y_i) \cdot \phi(y_j)}{\|\phi(y_i)\|\|\phi(y_j)\|}, \tag{24}$$
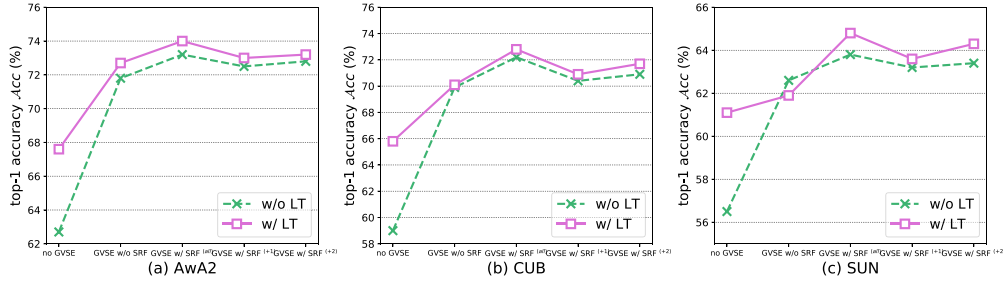
Fig. 9. Performance of different methods on AwA2, CUB, and SUN datasets with proposed split (PS) (%) in the inductive setting of CZSL tasks. Here, **no GVSE** indicates baseline model with inductive ResNet QFSL [7], and different settings of STF are compared. Moreover, **SRF**$^{(all)}$ denotes that the semantic features from all GCN blocks are merged with the visual embedding, and **SRF**$^{(+1)}$ and **SRF**$^{(+2)}$ indicate that only the last one or two GCN blocks are considered.

TABLE V
ABLATION RESULTS (%) WITH DIFFERENT TYPES OF SEMANTIC GRAPH AND DIFFERENT TYPES OF GRAPH MODELING STRATEGIES. FOR SEMANTIC GRAPH TYPE:
1. CATEGORY SIMILARITY GRAPH (CATE-BASED); 2. INSTANCE k-NN GRAPH FROM REFERENCE [68] (INS k-NN); 3. ATTRIBUTE CO-OCCURRENCE GRAPH IN THE
PROPOSED APPROACH (ATT-BASED). FOR SEMANTIC MODELING STRATEGY: 1. SERIAL MODE FOLLOW THE VISUAL MODELING (SERIAL); 2. THE PROPOSED
DUAL-PIPELINE ENTANGLEMENT STRUCTURE (DUAL-PIPE). ACCURACY ($\mathcal{ACC}$) IN CZSL WITH PROPOSED SPLIT (PS) IS RECORDED. RESULTS ARE RECORDED ON
DATASETS: AwA2, CUB, SUN

| | Model | Type of graph | | | Modeling strategy | AwA2 | CUB | SUN |
		Cate-based	Ins k-NN [68]	Att-based				
$\mathcal{I}$	ResNet-101	✓	–	–	Serial	68.8	68.7	59.9
		✓	–	–	Dual-pipe (ours)	69.4	69.5	60.2
		–	–	✓	Serial	69.9	70.3	62.0
		–	–	✓	Dual-pipe (ours)	**73.2**	**72.2**	**63.8**
$\mathcal{T}$	ResNet-101 + T	–	✓	–	Serial	82.2	77.1	61.0
		–	✓	–	Dual-pipe (ours)	82.0	76.8	61.9
		✓	–	–	Serial	81.4	76.0	60.8
		✓	–	–	Dual-pipe (ours)	81.8	76.5	61.4
		–	–	✓	Serial	82.4	77.6	61.3
		–	–	✓	Dual-pipe (ours)	**83.7**	**79.4**	**63.8**

where $\phi(y_i)$ and $\phi(y_j)$ are the attribute distributions to categories y_i and y_j. when $\mathcal{S}im(y_i, y_j) > 0.5$, we create an edge between categories y_i and y_j.

As listed in Table V, it can be seen that when the Category-based and Instance k-NN knowledge graphs are utilized, the model's classification performance is worse than using the Attribute-based graph, demonstrating that the Attribute-based graph proposed in the GVSE model can better suit the implicit semantic relations in the visual feature. Simultaneously, the proposed Dual-pipe architecture can also achieve better results than serial modeling after visual modeling, indicating that the optimization of graph semantic modeling is demanded in all visual layers of the hierarchy. **Appendix** records the complete ablation results on the type of semantic knowledge graph.

6) Analysis of Other Hyper-Parameters: We first conduct experiments to check the effect the PMI threshold δ, Table 10 lists the comparison ZSL results with different setting of δ, it can be seen that when $\delta = 0.75$, the GVSE-101 model get the best results on all datasets. And, when $\delta = 0.25$ and $\delta = 0.5$, their gaps to the best results reached $0.7\% \sim 2.4\%$ and $0.4\% \sim 1.7\%$ respectively. Moreover, in **Appendix**, we conduct more analysis of vertex and edge Information of the semantic knowledge graph at different δ settings.

Then, we present the results of using different word embedding tools for preparing the attribute word vector GCN sub-output. As listed in Table VII, when we use GloVe [55] and FastText [69], and a pre-computed word embedding library based on common-sense knowledge graphs: ConceptNet Numberbatch [70]. The ZSL performance of the model changes little

TABLE VI
COMPARISON RESULTS ON PROPOSED SPLIT (PS) (%) WITH DIFFERENT
SETUPS OF THE PMI THRESHOLD δ. THE EMBEDDING SUBNET IS GVSE-101

Model	PMI Threshold	AwA2	CUB	SUN
GVSE-101	$\delta = 0.25$	72.5	71.3	61.4
	$\delta = 0.50$	72.2	71.8	62.1
	$\delta = 0.75$	**73.2**	**72.2**	**63.8**

TABLE VII
COMPARISON RESULTS ON PROPOSED SPLIT (PS) (%) WITH DIFFERENT WORD
EMBEDDING TOOLS. THE EMBEDDING SUBNET IS GVSE-101

Model	Word Embedding Tool	AwA2	CUB	SUN
GVSE-101	Word2Vec [22]	73.2	72.2	**63.8**
	GloVe [55]	**73.6**	**72.4**	63.3
	FastText [69]	72.9	72.0	63.1
	ConceptNet [70]	72.3	70.8	62.0

on GloVe and FastText. GloVe brings the best outcomes on the AwA2 [18] and CUB datasets [19]. The model achieves the best result on the SUN dataset [20] with Word2Vec [22]. We do not choose a bigger language framework like BERT [71] because the amount of categories and attributes is tiny. In general, different self-training word embedding tools have few effects on GVSE's performance. However, when ConceptNet is utilized, the performance drops slightly. It should be noted that ConceptNet is a pre-computed word embedding package, we are only able to query attribute word, and obtain the word vector. The word vector based on common-sense knowledge graph brings more noise to ZSL image recognition in a specific field. Also, it cannot form self-consistent with the semantic graph modeling
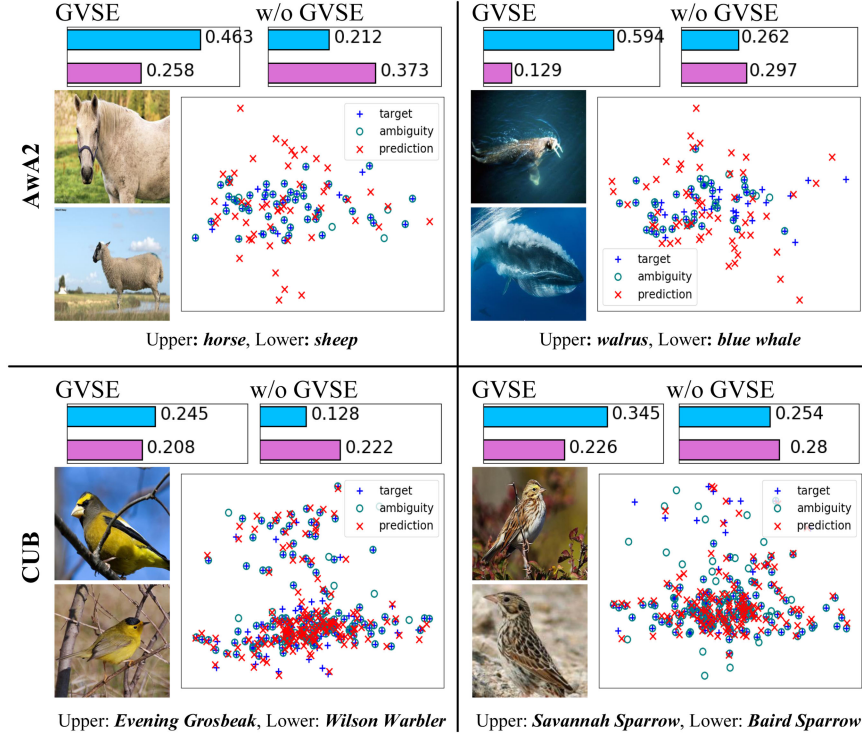
Fig. 10. Visualisation of the attribute word vector outcomes in CZSL task on AwA2 and CUB. The left side of each sub-figure are example images, the upper part is the input image, and the lower part is from the ambiguity class. The top of each sub-figure is the classification probabilities of the compared 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 GVSE, respectively. The PCA dimensionality reduction [73] is used.

TABLE VIII
COMPARISON RESULTS ON PROPOSED SPLIT (PS) (%) WITH DIFFERENT SRF FUSION STRATEGIES. THE GVSE-101 AND GVSE^{LT}-101 ARE USED AS THE EMBEDDING SUBNET

Model	SRF Fusion Strategy	AwA2	CUB	SUN
GVSE-101	Summation	72.6	71.9	**64.0**
	Concatenation	**73.2**	**72.2**	63.8
GVSE^{LT}-101	Summation	73.5	72.6	64.6
	Concatenation	**74.0**	**72.8**	**64.8**

based on attribute co-occurrence in GVSE. More comprehensive results and analysis on the comparison of word embedding tools are provided in **Appendix**.

Moreover, we discuss different SRF fusion methods. In addition to using the concatenation $\bowtie$ to combine SRF outcomes from different GCN block, we can also use the summation function (replace Eq. 14), as follows:

$$\theta(x)^+ = \theta(x) \bowtie \sum_{i=1}^{L} \hat{f}_{\mathcal{G}}^{(i)}, \quad (25)$$

Using the summation to fuse all SRF features can further reduce the width of the integrated visual embedding and reduce the computational cost of followed visual-semantic bridging.

We list the comparison results with different SRF fusion strategies in Table VIII, the concatenation method performs slightly better, and there is not much margin between the two. Among them, on the SUN dataset [20], the gap is smallest, and

sometimes the summation strategy is better, because the number of seen and unseen classes in the SUN dataset is hugely biased, and there are very few training samples in each category, and fewer parameters of the summation strategy can bring some benefits to generalization capabilities.

Appendix records more experimental results about hyper-parameter setting, include the setting of the balancing factor γ and more results with various word embedding tools for auxiliary output.

C. Generalized Comparison

GZSL test model's ability to recognize samples in seen and unseen mixed search space. We also verify the proposed GVSE network in the GZSL task. As listed in Table IX, we can notice that, except for a few classification results for seen classes, our proposed method outperforms other comparison methods in the most case on all datasets. For inductive methods, GVSE-101 is lower than TCN [5] in the AwA2 and SUN datasets. The reason is that although TCN does not access unseen samples when training, it utilizes the plentiful description semantic of the unseen class to help the training. For transductive methods, GVSE^{LT}-101 + SA surpasses state-of-the-art methods in harmonic score, and it improve over the corresponding baseline LFGAA + SA [16] by a significant margin of $6.4\% \sim 20.3\%$ on the AwA2, CUB, and SUN datasets. When using the transductive setting in [7], GVSE-101 + T achieves $2.1\% \sim 7.9\%$ improvement over the comparison methods. Although the forced

TABLE IX
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. $\mathcal{I}$: INDUCTIVE ZSL METHODS, AND $\mathcal{T}$: TRANSDUCTIVE ZSL METHODS

|	Method	AwA2			CUB			SUN		
		ACC^S	ACC^U	$\mathcal{H}$	ACC^S	ACC^U	$\mathcal{H}$	ACC^S	ACC^U	$\mathcal{H}$
$\mathcal{I}$	DEVISE [59]	**90.5**	10.0	18.0	**70.9**	11.5	19.8	43.3	7.9	13.4
	ESZSL [13]	77.8	5.9	11.0	63.8	12.6	21.0	27.9	11.0	15.8
	CMT* [72]	89.0	8.7	15.9	60.1	4.7	8.7	28.0	8.7	13.3
	CDL [61]	73.9	**29.3**	**41.9**	55.2	23.5	32.9	34.7	**21.5**	**26.5**
	TCN [5]	65.8	61.2	63.4	52.0	52.6	52.3	37.3	31.2	34.0
	LFGAA [16]	**93.4**	27.0	**41.9**	80.9	36.2	50.0	40.0	18.5	25.3
	GVSE-101 (ours)	94.0 ± 1.6	39.6 ± 2.1	55.7 ± 1.9	85.3 ± 1.3	38.8 ± 2.2	53.3 ± 1.9	39.8 ± 1.1	24.5 ± 1.3	30.3 ± 0.9
$\mathcal{T}$	AREN [3]	79.1	54.7	64.7	69.0	63.2	**66.0**	40.3	32.3	35.9
	SGMA [11]	87.1	37.6	52.5	71.3	36.7	48.5	–	–	–
	PREN [8]	88.6	32.4	47.4	55.8	35.2	27.2	35.4	27.2	30.8
	LFGAA + SA [16]	90.3	50.0	64.4	**79.6**	43.4	56.2	34.9	20.8	26.1
	HPL [32]	60.3	75.7	67.1	47.1	54.2	50.4	39.1	50.9	**44.2**
	APNet [36]	83.9	54.8	66.4	55.9	48.1	51.7	40.6	35.4	37.8
	E-PGN [28]	83.5	52.6	64.6	61.1	52.0	56.2	–	–	–
	DAZLE [31]	75.7	60.3	67.1	59.6	56.7	58.1	24.3	**52.3**	33.2
	Zero-VAE-GAN [30]	87.0 ± 3.6	**70.2 ± 3.2**	**77.6 ± 2.3**	57.9 ± 2.3	**64.1 ± 2.2**	60.8 ± 1.1	35.8 ± 1.0	**53.1 ± 1.4**	**42.8 ± 0.4**
	GVSE-50 + T (ours)	**90.8 ± 1.2**	65.4 ± 1.9	**76.0 ± 2.0**	69.2 ± 0.9	68.3 ± 1.1	**68.7 ± 1.3**	36.9 ± 1.8	50.6 ± 0.8	42.7 ± 1.7
	GVSE-101 + T (ours)	94.0 ± 1.3	69.2 ± 2.1	**79.7 ± 1.2**	74.8 ± 1.5	**73.1 ± 1.3**	**73.9 ± 1.3**	38.3 ± 1.2	**55.2 ± 1.7**	**45.2 ± 1.8**
	GVSE^{LT}-101 + SA (ours)	**93.1 ± 1.3**	57.1 ± 1.5	70.8 ± 2.0	80.6 ± 1.6	53.2 ± 1.3	64.1 ± 1.0	**43.3 ± 1.5**	31.4 ± 1.5	36.4 ± 1.4
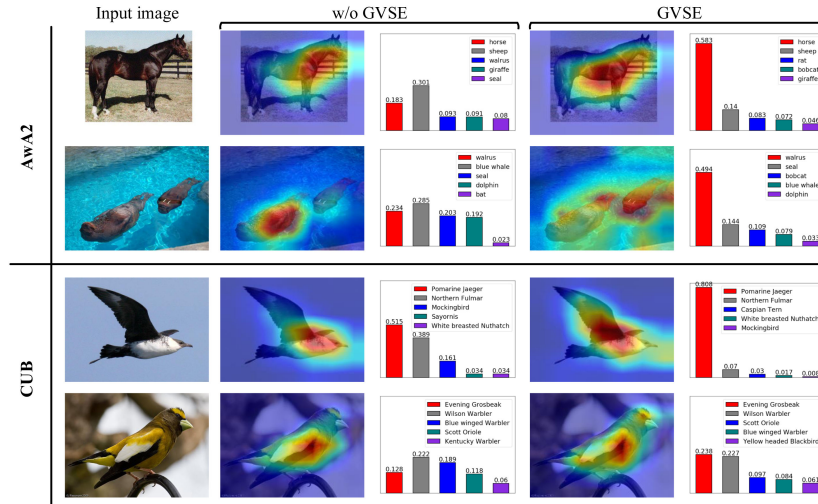
Fig. 11. The Grad-CAM visualization [74] and classification probabilities of GVSE^{LT}-101 and LFGAA (w/o GVSE). The **red** bar denotes the classification probability of target class, and the others indicate top-4 ambiguous classes.

reversal of unseen class weights [7] reduces the seen class accuracy on extremely biased datasets such as SUN, the GVSE-101 + T and GVSE-50 + T (with only a 50 layer backbone) still achieve the extremely competitive scores on the harmonic metric. Moreover, although many existing state-of-the-art methods utilize unseen images or semantics during training, they only achieve the best result on one or two datasets. In contrast, the GVSE network performs well on all datasets and can maintain an effective improvement when other methods fail.

Moreover, we also conduct the above ablation studies in Section V-B under the setting of GZSL, the results are provided in **Appendix**.

D. Visualization and Disambiguation Analysis

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

We apply inductive GVSE^{LT}-101 and LFGAA to compare the models with or without the GVSE, as shown by Fig. 10. Note that the target class and the ambiguity class share numerous attributes, and thus we can see that many symbols similar to '$\oplus$' indicate the coincidence of the attributes. For each dataset, the left and right sub-figures respectively indicate strong and slightly weak ambiguities. It can be seen that a different degree of ambiguity coincide directly with the classification probabilities, and the attribute word vector output by the GVSE 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 GVSE, whereas the model without an GVSE fails. For the ambiguity class with a lower coincidence, the GVSE also significantly reduces the interference of an ambiguous output.

The phenomenon shown in Fig. 11 confirms that there are implicit semantic correlations between the visual features in

various regions, which is in line with our expectations. The GVSE 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 GVSE is limited and incomplete. From the perspective of classification probability, the proposed GVSE successfully corrects and reduces the ambiguity.

VI. Conclusion

In this paper, we propose a Graph-based Visual-Semantic Entanglement 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, a CNN visual modeling pipeline and a GCN relationship graph modeling modeling pipeline parallel entangled network architecture is designed. 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 utilized.

In future works, We plan further to improve the proposed model from the following aspects: 1. Explore more open semantic knowledge graph construction methods and utilize dynamic graph GNN technology for relational semantic modeling; 2. Expand the proposed method to make it applicable to a wider-ranging data condition with the word vector as the attribute space; 3. The proposed method has a wide range of applicability, and we plan to collaborate it with more ZSL models further to verify its improvement effect on more ZSL models.

References

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

[2] E. Kodirov, T. Xiang, and S. Gong, "Semantic autoencoder for zero-shot learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3174–3183.

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

[4] M. Kampffmeyer, Y. Chen, X. Liang, H. Wang, Y. Zhang, and E. P. Xing, "Rethinking knowledge graph propagation for zero-shot learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 11 487–11 496.

[5] H. Jiang, R. Wang, S. Shan, and X. Chen, "Transferable contrastive network for generalized zero-shot learning," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 9765–9774.

[6] J. Ni, S. Zhang, and H. Xie, "Dual adversarial semantics-consistent network for generalized zero-shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 6143–6154.

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

[8] M. Ye and Y. Guo, "Progressive ensemble networks for zero-shot recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 11 728–11 736.

[9] Y. Li, J. Zhang, J. Zhang, and K. Huang, "Discriminative learning of latent features for zero-shot recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7463–7471.

[10] O. Russakovsky *et al.*, "ImageNet large scale visual recognition challenge," *Int. J Comput. vis.*, vol. 115, no. 3, pp. 211–252, 2015.

[11] Y. Zhu, J. Xie, Z. Tang, X. Peng, and A. Elgammal, "Semantic-guided multi-attention localization for zero-shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 14 917–14 927.

[12] Z. Zhang and V. Saligrama, "Zero-shot learning via semantic similarity embedding," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 4166–4174.

[13] B. Romera-Paredes and P. Torr, "An embarrassingly simple approach to zero-shot learning," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 2152–2161.

[14] V. Kumar Verma, G. Arora, A. Mishra, and P. Rai, "Generalized zero-shot learning via synthesized examples," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 4281–4289.

[15] Z. Wan *et al.*, "Transductive zero-shot learning with visual structure constraint," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 9972–9982.

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

[17] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*.

[18] Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata, "Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly," in *Proc. IEEE Trans. Pattern Anal. Mach. Intell.*, 2018.

[19] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The Caltech-UCSD Birds-200-2011 dataset." California Institute of Technology, Tech. Rep. CNS-TR-2011-001, 2011.

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

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

[22] 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.

[23] P. Morgado and N. Vasconcelos, "Semantically consistent regularization for zero-shot recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 6060–6069.

[24] Y. Fu, T. M. Hospedales, T. Xiang, and S. Gong, "Transductive multi-view zero-shot learning," *IEEE Trans. Pattern anal., Mach. Intell.*, vol. 37, no. 11, pp. 2332–2345, 2015.

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

[26] X. Xu, H. Cao, Y. Yang, E. Yang, and C. Deng, "Zero-shot metric learning," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, AAAI Press, 2019, pp. 3996–4002.

[27] L. Niu, J. Cai, A. Veeraraghavan, and L. Zhang, "Zero-shot learning via category-specific visual-semantic mapping and label refinement," *IEEE Trans. Image Process.*, vol. 28, pp. 965–979, Feb. 2019.

[28] Y. Yu, Z. Ji, J. Han, and Z. Zhang, "Episode-based prototype generating network for zero-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 14 0 35–14 044.

[29] Y. Xian, S. Sharma, B. Schiele, and Z. Akata, "F-VAEGAN-D2: A feature generating framework for any-shot learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 10 275–10 284.

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

[31] D. Huynh and E. Elhamifar, "Fine-grained generalized zero-shot learning via dense attribute-based attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 4483–4493.

[32] X. Zhang, S. Gui, S. Zhu, Y. Zhao, and J. Liu, "Hierarchical prototype learning for zero-shot recognition," *IEEE Trans. Multimedia*, vol. 22, no. 7, pp. 1692-1703, Jul. 2020.

[33] J. Guo and S. Guo, "A novel perspective to zero-shot learning: Towards an alignment of manifold structures via semantic feature expansion," *IEEE Trans. Multimedia*, vol. 23, pp. 524-537, Apr. 2021.

[34] X. Wang, Y. Ye, and A. Gupta, "Zero-shot recognition via semantic embeddings and knowledge graphs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6857–6866.

[35] J.-J. Chen, L. Pan, Z. Wei, X. Wang, C.-W. Ngo, and T.-S. Chua, "Zero-shot ingredient recognition by multi-relational graph convolutional network," in *Proc. 34th AAAI Conf. Artif. Intell.*, 2020.

[36] L. Liu, T. Zhou, G. Long, J. Jiang, and C. Zhang, "Attribute propagation network for graph zero-shot learning," in *Thirty-Fourth AAAI Conf. Artif. Intell.*, 2020.

[37] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, "Geometric deep learning: Going beyond euclidean data," *IEEE Signal Process. Mag.*, vol. 34, no. 4, pp. 18–42, Jul. 2017.

[38] T. Chen, L. Lin, R. Chen, Y. Wu, and X. Luo, "Knowledge-embedded representation learning for fine-grained image recognition," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, AAAI Press, 2018, pp. 627–634.

[39] R. Chen, T. Chen, X. Hui, H. Wu, G. Li, and L. Lin, "Knowledge graph transfer network for few-shot recognition," in *Proc. 34th AAAI Conf. Artif. Intell.*, 2020.

[40] L. Lu, Y. Lu, R. Yu, H. Di, L. Zhang, and S. Wang, "GAIM: Graph attention interaction model for collective activity recognition," *IEEE Trans. Multimedia*, vol. 22, no. 2, pp. 524–539, Feb. 2020.

[41] C. Sun, A. Shrivastava, C. Vondrick, R. Sukthankar, K. Murphy, and C. "Schmid, Relational action forecasting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 273–283.

[42] L. Zhang, X. Li, A. Arnab, K. Yang, Y. Tong, and P. H. Torr, "Dual graph convolutional network for semantic segmentation," in *Proc. 30th Brit. Mach. Vis. Conf.*, 2019.

[43] G. Li, M. Muller, A. Thabet, and B. Ghanem, "DeepGCNs: Can GCNs go as deep as CNNs?" in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 9267–9276.

[44] T. Chen, M. Xu, X. Hui, H. Wu, and L. Lin, "Learning semantic-specific graph representation for multi-label image recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 522–531.

[45] J. Gao and C. Xu, "Ci-gnn: Building a category-instance graph for zero-shot video classification," *IEEE Trans. Multimedia*, vol. 22, no. 12, pp. 3088-3100, Dec. 2020.

[46] P. Cui, S. Liu, and W. Zhu, "General knowledge embedded image representation learning," *IEEE Trans. Multimedia*, vol. 20, no. 1, pp. 198–207, Jan. 2018.

[47] K. Li, Y. Zhang, K. Li, Y. Li, and Y. Fu, "Visual semantic reasoning for image-text matching," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 4654–4662.

[48] J. Wang, W. Wang, L. Wang, Z. Wang, D. D. Feng, and T. Tan, "Learning visual relationship and context-aware attention for image captioning," *Pattern Recognit.*, vol. 98, 2020, Art. no. 107075.

[49] X. Li and S. Jiang, "Know more say less: Image captioning based on scene graphs," *IEEE Trans. Multimedia*, vol. 21, no. 8, pp. 2117–2130, Aug. 2019.

[50] K. Liu, L. Gao, N. M. Khan, L. Qi, and L. Guan, "A multi-stream graph convolutional networks-hidden conditional random field model for skeleton-based action recognition," *IEEE Trans. Multimedia*, vol. 23, pp. 64-76, 2021.

[51] G. Bouma, "Normalized (pointwise) mutual information in collocation extraction," in *Proc. GSCL*, 2009, pp. 31–40.

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

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

[54] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 7794–7803.

[55] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proc. 2014 Conf. Empirical methods Natural Lang. Process.*, 2014, pp. 1532–1543.

[56] D. Huynh and E. Elhamifar, "A shared multi-attention framework for multi-label zero-shot learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 8776–8786.

[57] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2117–2125.

[58] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning Representations by Back-Propagating Errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[59] A. Frome *et al.*, "Devise: A deep visual-semantic embedding model," in *Adv. Neural Inf. Process. Syst.*, 2013, pp. 2121–2129.

[60] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid, "Label-embedding for attribute-based classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2013, pp. 819–826.

[61] H. Jiang, R. Wang, S. Shan, and X. Chen, "Learning class prototypes via structure alignment for zero-shot recognition," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 118–134.

[62] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 815–823.

[63] M. Cuturi, "Sinkhorn distances: Lightspeed computation of optimal transport," in *Adv. Neural Inf. Process. Syst.*, 2013, pp. 2292–2300.

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

[65] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv:1412.6980*, 2014.

[66] L. Chen *et al.*, "SCA-CNN: Spatial and channel-wise attention in convolutional networks for image captioning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 5659–5667.

[67] S. Woo, J. Park, J.-Y. Lee, and I. So Kweon, "Cbam: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19.

[68] M. Rohrbach, S. Ebert, and B. Schiele, "Transfer learning in a transductive setting," in *Adv. Neural Inf. Process. Syst.*, 2013, pp. 46–54.

[69] 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.

[70] R. Speer, J. Chin, and C. Havasi, "Conceptnet 5.5: An open multilingual graph of general knowledge," in *Proc. the AAAI Conf. Artif. Intell.*, vol. 31, no. 1, 2017.

[71] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proc. 2019 Conf. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, Vol. 1 (Long and Short Papers), 2019, pp. 4171–4186.

[72] R. Socher, M. Ganjoo, C. D. Manning, and A. Ng, "Zero-shot learning through cross-modal transfer," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 935–943.

[73] M. E. Tipping and C. M. Bishop, "Probabilistic principal component analysis," *J. Roy. Stat. Society: Ser. B. (Statistical Methodol.)*, vol. 61, no. 3, pp. 611–622, 1999.

[74] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Conf. Comput. Vis.*, 2017, pp. 618–626.

Yang Hu (Student Member, IEEE) received the M.A.Eng. degree, in 2016 from the Kunming University of Science and Technology, Kunming, China, where he is currently working the Ph.D. degree with the South China University of Technology, Guangzhou, China. He is also a Visiting Researcher with Web Science Institute, University of Southampton, Southampton, U.K. His research interests include neural network and deep learning, knowledge graph, and biomedical information processing.

Guihua Wen received the Ph.D. degree in computer science and engineering, South China University of Technology, Guangzhou, China. He is currently a Professor and Doctoral Supervisor with the School of Computer Science and Technology, South China University of Technology. He is also Professor-in-chief of the Data Mining and Machine Learning Laboratory, South China University of Technology. His research interests include cognitive affective computing, machine learning, and data mining.

Adriane Chapman received the Ph.D. degree from the University of Michigan, Ann Arbor, MI, USA, in 2008. She is currently an Associate Professor of electronics and computer science with the University of Southampton, Southampton, U.K. She joined The MITRE Corporation, McLean, VA, USA, a non-profit Federally Funded Research and Development Company that researched and applied solutions to technology problems in the U.S. government. In 2016, she joined the University of Southampton. She is the Co-Director of the ECS Centre for Health Technologies, and is passionate about applying technology for society's benefit. She is currently working on the following projects, digital predictive technologies and conduits of algorithmic bias in policing and penal services, enabling personalized expression of privacy requirements.

Pei Yang (Member, IEEE) received the Ph.D. degree from the South China University of Technology, Guangzhou, China. He is currently an Associate Professor with the South China University of Technology. He is conducting postdoctoral research with Arizona State University, Tempe, AZ, USA. He has authored or coauthored more than 30 research articles on referred journals and conference proceedings, including IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, *ACM Transactions on Knowledge Discovery from Data*, KDD, ICDM, IJCAI, ACL, and WSDM. His research focuses on statistical machine learning and data mining, including heterogeneous learning, semi-supervised learning, transfer learning, rare category analysis, and functional data analysis, with applications in web mining, big data analytics, bioinformatics, and healthcare. He was on the program committee of conferences, including NIPS, ICML, KDD, ICDM, SDM, IJCAI, and AAAI.

Mingnan Luo is currently working toward the master's degree with the College of Computer Science and Engineering, South China University of Technology, Guangzhou, China. His main research interests include image processing and deep learning.

Yingxue Xu is currently working toward the master's degree with the College of Computer Science and Engineering, South China University of Technology, Guangzhou, China. Her main research interests include deep learning and medical image processing.

Dan Dai received the M.A.Eng. degree from the Kunming University of Science and Technology, Kunming, China, in 2016 and the Ph.D. degree from the South China University of Technology, Guangzhou, China. She is currently a Postdoctor Research Fellow with the University of Lincoln, Lincoln, U.K. Her research interests include machine learning, computer vision, and robot technology.

Wendy Hall received the D.B.E, F.R.S, and F.R.Eng. degrees. She is currently a Regius Professor of computer science with the University of Southampton, Southampton, U.K., and the Executive Director of the Web Science Institute. In 2009, she became a Dame Commander of the British Empire, U.K. New Year's Honours list, and is a Fellow of the Royal Society. She has previously been President of the ACM, Senior Vice President of the Royal Academy of Engineering, and a Member of the U.K. Prime Minister's Council for Science and Technology. She is also the Co-Chair of the U.K. government's AI Review.