(ML)^{2}P-Encoder: On Exploration of Channel-class Correlation for Multi-label Zero-shot Learning

Ziming Liu[1], Song Guo[1,2], Xiaocheng Lu[1], Jingcai Guo[1,2]*, Jiewei Zhang[1], Yue Zeng[1], Fushuo Huo[1]
[1]Department of Computing, The Hong Kong Polytechnic University, Hong Kong SAR, China
[2]The Hong Kong Polytechnic University Shenzhen Research Institute, Shenzhen, China
{ziming.liu, jiewei.zhang, fushuo.huo}@connect.polyu.hk
{song.guo, xiaoclu, jc-jingcai.guo, zengyue.zeng}@polyu.edu.hk

Abstract

*Recent studies usually approach multi-label zero-shot learning (MLZSL) with visual-semantic mapping on spatial-class correlation, which can be computationally costly, and worse still, fails to capture fine-grained class-specific semantics. We observe that different channels may usually have different sensitivities on classes, which can correspond to specific semantics. Such an intrinsic channel-class correlation suggests a potential alternative for the more accurate and class-harmonious feature representations. In this paper, our interest is to fully explore the power of channel-class correlation as the unique base for MLZSL. Specifically, we propose a light yet efficient **M**ulti-**L**abel **M**ulti-**L**ayer **P**erceptron-based **Encoder**, dubbed (ML)^{2}P-Encoder, to extract and preserve channel-wise semantics. We reorganize the generated feature maps into several groups, of which each of them can be trained independently with (ML)^{2}P-Encoder. On top of that, a global group-wise attention module is further designed to build the multi-label specific class relationships among different classes, which eventually fulfills a novel **C**hannel-**C**lass **C**orrelation MLZSL framework (C^3-MLZSL)[1]. Extensive experiments on large-scale MLZSL benchmarks including NUS-WIDE and Open-Images-V4 demonstrate the superiority of our model against other representative state-of-the-art models.*

1. Introduction

The proliferation of smart devices has greatly enriched human life when it comes to the era of big data. These smart devices are usually equipped with cameras such that users can easily produce and share their images. With the increasing abundance of public images, how to analyze them accurately has become a challenging problem. Recent years have witnessed great success in classifying an image into a specific class [20, 37, 39], namely, single-label classification. However, in reality, the images [17,46] usually contain abundant information and thereby consist of multiple labels.

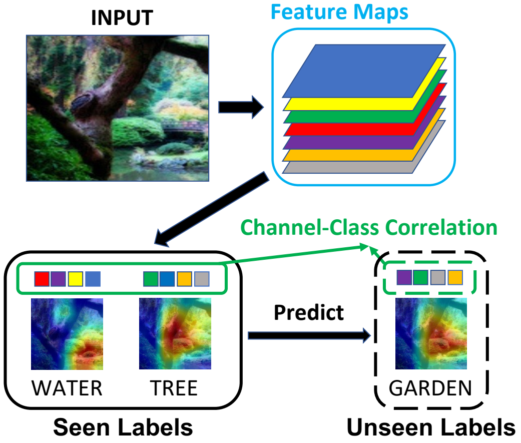


Figure 1. **Example of Channel-Class Correlation.** Our method achieves the prediction of unseen classes by exploiting the unique distribution of channel responses as semantic information for the class and building correlations with responses from the same channel (zoom in for a better view).

In recent years, the multi-label classification has been widely investigated by exploring the relationship among different labels from multiple aspects [9, 13, 14, 16, 42]. However, in some scenarios where extensive collections of images exist, e.g., Flickr[2], users can freely set one or more individual tags/labels for each image, while the presented objects and labels in these images may not be fully shown in any previous collection, and thus result in a domain gap for the recognition. Therefore, in real-world applications, the model is required to gain the ability to predict unseen classes as well. As one of the thriving research topics, zero-

*Jingcai Guo is the corresponding author.
[1] Released code: github.com/simonzmliu/cvpr23_mlzsl

[2] https://www.flickr.com

shot learning (ZSL) [1, 12, 15, 34] is designed to transfer tasks from seen classes to unseen classes, and naturally recognizes novel objects of unseen classes. Specifically, ZSL has made continuous success in single-label classification [19, 26, 31, 45, 48]. However, these methods can hardly be extended to the multi-label scenario since exploring the cross-class relationships in an image is non-trivial.

Recently, some works have focused on multi-label zero-shot learning (MLZSL) tasks and obtained some promising results [33, 36, 49]. Other works considered incorporating attention mechanisms into their models, such as *LESA* [22] and *BiAM* [35]. *LESA* [22] designed an attention-sharing mechanism for different patches in the image so that each patch can output the corresponding class. In another way, *BiAM* [35] designed a bi-level attention to extract relations from regional context and scene context, which can enrich the regional features of the model and separate the features of different classes.

Although previous works have made considerable progress, their designed methods have been limited to the processing of spatial-domain information. First of all, the over-reliance on spatial-class correlation fails to capture fine-grained class-specific semantics. In addition, the additional processing of spatial information greatly increases the computational cost of the model and limits the inference speed. Given the shortcomings of the above methods, we found through analysis that the channel response can be used as the semantic information of the class. Firstly, the response of each class in the channel is unique, which creates conditions for obtaining the unique semantics. Secondly, for classes with certain semantic associations, there must be some channels that capture their common information. Therefore, channel information, as an easily overlooked part after feature extraction, can complete the task of capturing multi-label information. In MLZSL, we can complete the prediction of unseen classes by obtaining the responses of seen classes in the channel domain, and the relationship between seen and unseen classes. Finally, the subsequent analysis of the channel response greatly saves computational costs.

Specifically, as shown in Figure 1, as seen classes, "water" and "tree" have unique response distributions on feature channels, and these responses can be used as semantic information for classification tasks. Besides, in order to explore the correlation of classes, we found that although the semantic information of "water" and "tree" is different, there are still some channels that respond simultaneously (i.e. the blue channel). We need to build this correlation during the training process through modeling so that the model can learn multi-label correlations. In the ZSL process, for the unseen class "garden", we know that it is related to "water" (i.e. purple layer) and "tree" (i.e. green, orange, and gray layer) by obtaining its semantic information and matching with seen classes. This observation suggests that channels can help not only to classify objects but also to establish associations between classes. Previous methods which only consider spatial information are unable to obtain this intrinsic channel-class correlation and dissimilarity, thus achieving sub-optimal performance on the MLZSL task.

To address the above challenges and construct a more accurate and robust MLZSL system, we propose to group the generated feature maps and process them in a group-wise manner, thus enhancing the model by fully exploring the channel-class correlations. Besides, by properly designing a light yet efficient Multi-Label Multi-Layer Perceptron-based Encoder, i.e., (ML)^{2}P-Encoder, we can easily analyze the local relationship between channels while significantly reducing the computation overhead. Finally, these groups are recombined and then perform the calculation of group attention, indicating that the model is analyzed locally and globally from the perspective of the channels, which can ensure the integrity of the representation.

In summary, our contributions are four-fold:

1. To the best of our knowledge, our method first suggests the concept of channel-class correlation in MLZSL, and proposes a channel-sensitive attention module (ML)^{2}P-Encoder to extract and preserve channel-wise semantics for channel groups.

2. Different from previous works that use spatial-class correlation to extract global and local features, we alternatively explore the channel-class correlation as the unique base for MLZSL.

3. In conjunction with (ML)^{2}P-Encoder, a global group-wise attention is also designed to establish the multi-label specific class relationships among classes.

4. Extensive experiments on large-scale datasets *NUS-WIDE* and *Open-Images*-V4 demonstrate the effectiveness of our method against other state-of-the-art models.

2. Related Work

2.1. Multi-Label Classification

The establishment of graph neural networks (GNNs) brings remarkable success to multi-label classification tasks [8, 25]. Among them, Chen *et al.* [8] constructs directed graphs for object labels and uses graph convolutional networks (GCN) to map label nodes, which contain word embeddings, into classifiers. In addition, the CNN-based multi-label classification models enable the learning of the characteristics of each label from the spatial information of the image and design a new multi-label classifier [13, 14, 16, 17, 42, 43, 46]. Gao *et al.* [16] suggests a two-stream framework to identify global and local information

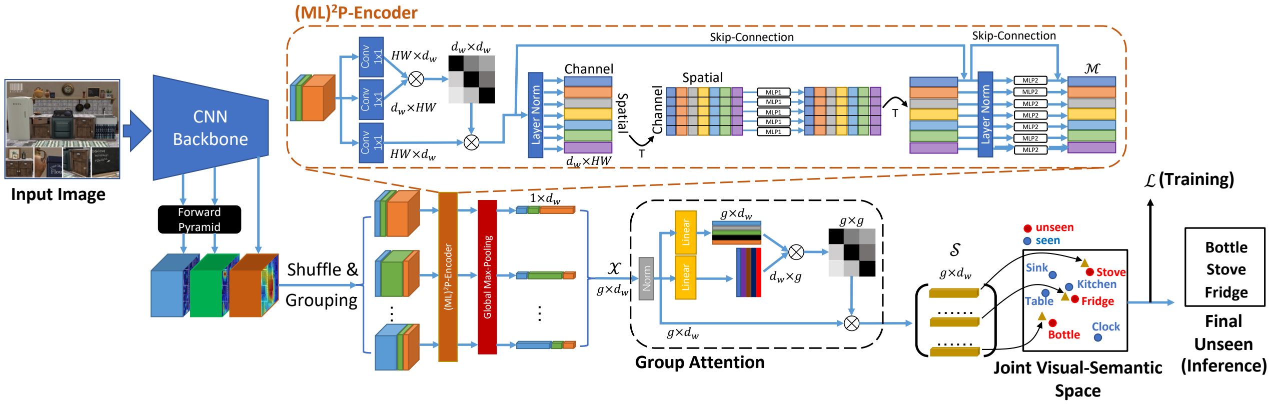


Figure 2. **Pipeline for C^3-MLZSL.** The input image is first passed through the feature extraction network (eg. VGG19), and then multi-layer feature maps are extracted through the **Forward Pyramid** module. After the feature maps are shuffled and grouped, each group uses **(ML)^{2}P-Encoder** to extract semantic information. Then, the semantic information generated by all groups is associated through **Group Attention** to generate the final semantic matrix $\mathcal{S}$ (zoom in for a better view).

separately and a multi-class regional attention module to align them. However, the above methods cannot generalize to unseen classes.

2.2. Zero-Shot Learning

Zero-shot learning provides a solution to recognize unseen classes. Current studies mostly consider a relatively simple single-label scenario [4, 6, 26, 30, 32, 47, 50, 51]. In practice, existing methods usually focus on finding the main semantic information of training images, and then exploit the semantic relationship, i.e., word vectors [15, 38, 44, 45] or attribute vectors [3, 27, 28], between seen and unseen classes for prediction. The generated semantic information can be inferred from seen to unseen labels by comparing the similarity of the relation vectors between them. For example, Chen *et al.* [7] proposes a generative flow framework and uses a combinatorial strategy to solve the problems of semantic inconsistency, variance collapse, and structural disorder in zero-shot learning. Gune *et al.* [18] generates visual proxy samples to simulate the average entropy of the label distribution of the unseen class. However, the above methods only predict single labels with a single representation of images, which can hardly generalize to a more realistic multi-label scenario.

2.3. Multi-Label Zero-Shot Learning

Multi-label zero-shot learning has received increasing attention recently. For example, Norouzi *et al.* [36] designs two separate spaces, i.e., the image and semantic embedding spaces, jointly with the convex combination of the label embedding vectors to achieve multi-label recognition in the zero-shot learning framework. Zhang *et al.* [49] proposes a fast and general model based on the fact that the word vectors of the relevant labels are ranked before the irrelevant word vectors in the main vector of the image. Different from the above methods, Lee *et al.* [29] uses the knowledge graph to connect different labels. In recent years, attention-based methods become the mainstream. For example, *LESA* [22] applies an attention-sharing mechanism to the multi-label environment, allowing the model to focus on the key areas of each label. Narayan *et al.* [35] uses a bi-layer attention module to combine global context information and local features and map the generated information to the semantic space. However, the above methods only stay at the two-dimensional space level ($H \times W$), and do not consider the response between different feature channels with respect to classes.

3. Methods

3.1. Problem Setting

Before proposing our method, we first explain the definition of the MLZSL problem. Given n input samples $\{(I_1, Y_1), \ldots, (I_i, Y_i), \ldots, (I_n, Y_n)\}$, where I_i represents the input image of the i-th train-set, and Y_i represents the training labels corresponding to the input images, which are also called 'seen labels'. On the label distribution, let us set the seen label in the dataset as $\mathcal{C}_s$, where the seen label refers to the label known by the model. $\mathcal{C}_s$ is mainly used for the train-set of the model in zero-shot learning. We set the unseen label to $\mathcal{C}_u$, and the unseen label is generally used in the test-set. The label relationship in the dataset is defined as $\mathcal{C} = \mathcal{C}_s \cup \mathcal{C}_u$, where $\mathcal{C}$ represents the set of all labels in the dataset. Based on the above definition, after the model is trained on the train-set, in the testing part of MLZSL, given the image I_u, the model can output the prediction result $y_u \subset \mathcal{C}_u$. While in the generalized zero-shot learning task, given an image I_u, the output of the model is

$y_u \subset \mathcal{C}$, which means the model needs to output both the seen label and the unseen label that exist in the image.

3.2. (ML)^{2}P-Encoder

The proposed network structure is shown in Figure 2. For input images I, we first use a pre-trained feature extraction network to obtain the corresponding image features $\mathcal{F}$. We extract the features from the last three layers of the feature extraction network, and keep the two layers with the larger size consistent with the smallest size layer by downsampling. For example, assuming that the used and training network is VGG19 [37], the size of the last three layers of feature maps is $\{28 \times 28, 14 \times 14, 7 \times 7\}$. We use max-pooling to down-sample the large-scale feature maps to obtain equivalent 7×7 feature maps. This step is called the "Forward Pyramid". After that, we obtain feature maps at different levels with the same scale. Then we randomly shuffle them to get the feature map $\mathcal{F}_a$ and re-group them into g different groups, each group has d_w channels, which is the same length as the word vectors in the ground-truth semantic space. The purpose of this operation is to generate specific semantic vectors to express the semantic information contained in each group.

Next, the features of each group are fed into (ML)^{2}P-Encoder. First, we need to calculate the correlation between channels within each group. In traditional self-attention, the cost of computation greatly consumes the inference speed of the model, and the traditional self-attention module cannot accurately reflect the relationship between each channel. To solve the loss caused by the amount of calculation and accurately reflect the channel correlation, we designed a new self-attention structure to achieve this.

For features $\mathcal{F}_a$ in group i, which is $\mathcal{F}_a^i \in \mathbb{R}^{H \times W \times d_w}$. We first generate Query ($\mathbf{Q}$), Value ($\mathbf{V}$) and Key ($\mathbf{K}$) through three convolution operations:

$$\mathbf{Q} = W_p^Q \mathcal{F}_a^i \quad \mathbf{K} = W_p^K \mathcal{F}_a^i \quad \mathbf{V} = W_p^V \mathcal{F}_a^i \tag{1}$$

where $W_p^{(\cdot)}$ means the convolution operation. Next, to obtain the channel correlation matrix $\mathcal{R}$, we reshape $\mathbf{Q}$, $\mathbf{K}$ and $\mathbf{V}$ in the spatial domain ($H \times W$) to get $\widehat{\mathbf{Q}} \in \mathbb{R}^{HW \times d_w}$, $\widehat{\mathbf{K}} \in \mathbb{R}^{d_w \times HW}$ and $\widehat{\mathbf{V}} \in \mathbb{R}^{HW \times d_w}$. Then perform a dot product operation on $\mathbf{Q}$ and $\mathbf{K}$ to obtain the channel correlation matrix $\mathcal{R} \in \mathbb{R}^{d_w \times d_w}$. After that, we do the dot product between $\mathcal{R}$ and $\mathbf{V}$, finally, add with the input $\mathcal{F}_a^i$ to get the output $\widehat{\mathcal{F}}_a^i \in \mathbb{R}^{H \times W \times d_w}$:

$$\text{Att}(\widehat{\mathbf{Q}}, \widehat{\mathbf{K}}, \widehat{\mathbf{V}}) = \widehat{\mathbf{V}} \cdot \text{softmax}(\underbrace{\widehat{\mathbf{K}} \cdot \widehat{\mathbf{Q}}}_{\mathcal{R}}) \tag{2}$$

$$\widehat{\mathcal{F}}_a^i = \mathcal{F}_a^i + \text{Att}(\widehat{\mathbf{Q}}, \widehat{\mathbf{K}}, \widehat{\mathbf{V}}) \tag{3}$$

After enhancing the correlation between channels, we need to extract and analyze the feature information contained in each channel. We reshape the information in the spatial domain into a one-dimensional vector, then we decide to use the Multi-Layer Perceptron (MLP) to encode the features. Compared with the traditional convolution structure, the MLP structure is convenient to perform information fusion between local regions. Specifically, for the input feature $\widehat{\mathcal{F}}_a^i \in \mathbb{R}^{H \times W \times d_w}$, we first change the dimension from $H \times W \times d_w$ to $\mathcal{F}_{mlp}^i \in \mathbb{R}^{d_w \times HW}$, then we use LayerNorm to normalize the input. Our MLP structure includes two different MLPs: MLP1 is used to extract the spatial information contained in each channel, and MLP2 is proposed to obtain local information of different channels in the spatial domain:

$$\mathcal{F}_{mlp1}^i = \mathcal{F}_{mlp}^i + \mathbf{W}_2 \sigma(\mathbf{W}_1 \text{LayerNorm}(\mathcal{F}_{mlp}^i)) \tag{4}$$

$$\mathcal{M} = \mathcal{F}_{mlp1}^i + \mathbf{W}_4 \sigma(\mathbf{W}_3 \text{LayerNorm}(\mathcal{F}_{mlp1}^i)) \tag{5}$$

where $\mathcal{F}_{mlp1}^i$ is the output after MLP1. $\mathbf{W}_1$, $\mathbf{W}_2$ is the parameter of MLP1, and $\mathbf{W}_3$, $\mathbf{W}_4$ is the parameter of MLP2. σ is an element-wise non-linearity GELU [21]. Then we use max-pooling to filter out the best semantic vector in the spatial domain, which can more accurately represent the semantic information of this group. This max-pooling operation is also to be able to directly extract the channel response. So we obtain group semantic vectors $\mathcal{X} \in \mathbb{R}^{g \times d_w}$ and send them into Group Attention.

3.3. Group Attention

Although we obtained group semantic vectors $\mathcal{X}$ through (ML)^{2}P-Encoder, the semantic vectors generated by each group did not establish a relationship with each other at this time. As we already know, the key to improving the accuracy of multi-label image classification is to construct the correlation of labels within the image. So we use Group Attention to build the mutual information and also to find similar responses between different labels. We pass a series of linear layers to $\mathcal{X}$:

$$\mathbf{Q_x} = W_x^Q \mathcal{X} \quad \mathbf{K_x} = W_x^K \mathcal{X} \tag{6}$$

$$\mathcal{S} = (\mathbf{Q_x} \cdot \mathbf{K_x}) \cdot \mathcal{X} \tag{7}$$

where $\mathbf{Q_x} \in \mathbb{R}^{g \times d_w}$, and we transpose $\mathbf{K_x}$ into $\mathbf{K_x} \in \mathbb{R}^{d_w \times g}$. W_x^Q and W_x^K are different linear weights. $\mathcal{S} \in \mathbb{R}^{g \times d_w}$ is the semantic matrix, which contains all the semantic information of the input image. In the loss function, we will make each semantic vector in $\mathcal{S}$ approximate the semantic information of seen classes appearing in the image. Therefore, from another perspective, the semantic vectors in $\mathcal{S}$ are related to seen classes.

3.4. Loss Function

During training, some semantic vectors are generated for each input image. The semantic matrix $\mathcal{S}$ includes the semantic information in the image and is sent to the prediction module. The loss function consists of two parts. First of all, to make the positive class (seen class appear in each training image) get a higher ranking than the negative class (seen class which does not appear in the training image). Inspired by [49], we choose to adopt ranknet loss [5] as the main component of the loss function. We use

$$\mu_{ij} = \max\left(\mathcal{S} \cdot n_i\right) - \max\left(\mathcal{S} \cdot p_j\right), \tag{8}$$

to indicate the number of violations of any of these ranking constraints, where n_i represents the semantic vector of the negative class, and p_j denotes the semantic vector of the positive class. $\max$ is used to maximize this gap between negative and positive, and constrain it in subsequent steps.

Next, to minimize the gap, we design the loss function as the following:

$$\mathcal{L}_{rank} = \frac{1}{\left(|P|\left|\bar{P}\right|\right)} \sum_i \sum_j \log\left(1 + e^{\mu_{ij}}\right), \tag{9}$$

where $\frac{1}{\left(|P|\left|\bar{P}\right|\right)}$ is used to normalize the ranknet loss, and $|P|$ denotes the number of positive class, $\left|\bar{P}\right|$ represents the number of negative class. When an image contains a large number of positive labels, the image becomes difficult to classify. So we need the model to value these hard samples during training. Therefore, we add the class weight ω to improve the performance of the model in the face of hard samples. ω is represented as:

$$\omega = 1 + \sum_i var(P^i), \tag{10}$$

where P^i represents the vector of the i-th positive class, var means the variance. The higher ω means the image contains more complex labels. To prevent the direction of the semantic vectors generated by the model from being too divergent, it needs to be controlled by the loss function. Therefore, we believe that the addition of regularization terms can reduce the difference between the generated semantic vectors when the model faces complex input images. This reduction in variance helps the model learn relevant information between different classes.

$$\mathcal{L}_{reg} = \left\| \sum_n var(\mathcal{S}_n) \right\|_1. \tag{11}$$

Finally, the loss function of the model is defined as:

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} \left((1-\lambda) \cdot \omega \mathcal{L}_{rank}(\mathcal{S}_i, Y_i) + \lambda \mathcal{L}_{reg}(\mathcal{S}_i)\right), \tag{12}$$

where N means the number of batch size, and λ is a hyperparameter that denotes the regularization term's weight.

4. Experiments

4.1. Experimental Setup

Datasets: First, we use the *NUS-WIDE* dataset [10] to conduct MLZSL experiments. The *NUS-WIDE* dataset contains about 270,000 images, and each image contains 925 labels, which are automatically extracted from Flickr user tags. In addition, it also contains 81 labels that are manually annotated by humans, and these labels are called 'Ground-Truth'. During the experiment, 925 labels were used as 'seen labels', and 81 labels were used as 'unseen labels'. This setting is similar with [22]. Another dataset is called the *Open-Images*-V4 dataset. This dataset contains nearly 9 million training images, 125,456 images as test images, and 41,620 images in the validation set. The train-set contains 7,186 labels, which are 'seen labels' that appear at least 100 times in the train-set. While the remaining 400 most frequent labels that do not appear in the train-set are used as test-set labels, they are also used as 'unseen labels'. Each unseen label has at least appeared 75 times.

Evaluation Metrics: To better allow our proposed new model and other comparative models to perform an unbiased comparison on the task of MLZSL, we use the two most common evaluation metrics, the mean Average Precision (mAP) [22, 41] and F1-Score. Among them, *top-K* F1-Score is used to measure the accuracy of the model for label prediction, and mAP is used to reflect the accuracy for unseen label retrieval of the image.

Implementation Details: Our model can support end-to-end training. We choose VGG19 [37], pre-trained on *ImageNet* dataset [11], as the backbone network. Unlike other methods, our model uses multi-scale feature maps and aggregates them. The sizes of the feature maps are 28 × 28, 14 × 14, and 7 × 7, respectively.

In terms of the optimizer, we choose to use the Adam optimizer [24], which requires less memory and is suitable for large datasets. The weight decay of the Adam optimizer is set to $4e^{-3}$. In the *NUS-WIDE* dataset experiments, the initial learning rate of the model is $5e^{-5}$, and then the learning rate decreases by $\frac{1}{10}$ at the 7th epoch. The entire experimental process of the *NUS-WIDE* dataset requires a total of 20 epochs with a batch size of 48. In the experiments using the *Open-Images*-V4 dataset, our learning rate, batch size, and decay rate remain the same as the *NUS-WIDE* dataset, but the number of epochs is 7.

Baselines: We will compare the proposed method with several state-of-the-art deep learning-based MLZSL models. These comparative methods have been published in recent years and cover a fairly rich variety of techniques, such as the attention mechanism with the most common CNNs. These comparison methods include: *CONSE* [36], *LabelEM* [2], *Fast0Tag* [49], Kim *et al.* [23], *LESA Attention per Cluster (ApC)* [22], *LESA* [22], and *BiAM* [35]. All comparison methods using VGG19 [37] are not fine-tuned. In addition to comparing with comparison models, we will also test the model's performance under different settings of hyper-parameters g and λ. At the same time, we will conduct ablation experiments to verify the integrity of the model's architecture.

4.2. State-of-the-art Comparison

NUS-WIDE: Table 1 shows the performance of ours and competitive methods on the *NUS-WIDE* test-set. The table contains the results of both ZSL and GZSL. *CONSE* [36] and *LabelEM* [2], as the methods proposed earlier, do not perform well on large-scale datasets. *Fast0Tag* [49] achieves more competitive results by sorting the positive labels to find the principal directions of the image. *LESA* [22] and *BiAM* [35] are currently the most advanced models that rely on spatial attention mechanism to generate semantic information. Compared to *BiAM*, our method achieves a 3.6% improvement on mAP in the ZSL task. Besides, we lead *BiAM* by 0.8% and 2.9% in F1-Score of $K = 3$ and $K = 5$, respectively. On the GZSL task, we also surpass *BiAM*. *BiAM* deals with higher-dimensional and richer spatial information, while our method is more inclined to single-dimensional channel responses. Therefore, it is not easy to achieve such results with 1.3% improvement in mAP and 0.3% and 0.7% in F1-Score of $K = 3$ and $K = 5$, respectively. Good results on *NUS-WIDE* dataset imply the effectiveness of our method.

Attention Visualization on NUS-WIDE: Figure 6 illustrate the attention regions of the model when our method predicts unseen labels. Figure 6(a) shows that our model can clearly distinguish scene information from all unseen classes. The attention areas of "Rocks" and "Mountain" in the figure are roughly the same, which indicates that the two classes have similar semantics and dependencies, and the existence of Group Attention enables the model to learn this mutual information well. Figure 6(b) is a comparison with BiAM [35], the best existing model for mining spatial domain information. This result fully shows the effective use of channel information can more accurately grasp the response between classes. While BiAM's over-exploration of spatial information improves the acquisition of regional information, it loses the scene-level response at the same time. For more comparison results, please refer to appendix.

Open-Images-V4: From Table 2, we show the results of ours and the baseline models on *Open-Images*-V4. We follow the evaluation setting of [22, 35]. This dataset contains more seen and unseen labels than *NUS-WIDE*. With a large increase in the number of classes, all methods get poor F1-Score on the ZSL task. Among them, *Fast0Tag* has made great progress compared with past methods, especially in the GZSL task. *LESA* [22] and *BiAM* [35], as the two best methods, represent the highest level of extracting spatial responses. *BiAM* achieves very large progress in mAP metrics on both ZSL and GZSL tasks. But our method achieves the best results in the mAP of ZSL, while leading by 3.4% and 2.8% in F1-Score with $K = 3$ and $K = 5$, respectively. Most importantly, for the GZSL task, our F1-Score results also achieve huge advantages by 10.0% and 9.0%. This shows that the channel-class correlation as semantic information can fully cope with the complex situation of a large number of labels.

Table 1. State-of-the-art comparison for **multi-label ZSL and GZSL** tasks on the **NUS-WIDE** dataset. We show the indicators of F1-Score in the case of $K \in 3, 5$ and mAP. The best results are shown in bold.

Method	**Task**	**mAP**	**F1 (K = 3)**	**F1 (K = 5)**
CONSE [36]	ZSL	9.4	21.6	20.2
	GZSL	2.1	7.0	8.1
LabelEM [2]	ZSL	7.1	19.2	19.5
	GZSL	2.2	9.5	11.3
Fast0Tag [49]	ZSL	15.1	27.8	26.4
	GZSL	3.7	11.5	13.5
Kim *et al.* [23]	ZSL	10.4	25.8	23.6
	GZSL	3.7	10.9	13.2
Attention per Cluster [22]	ZSL	12.9	24.6	22.9
	GZSL	2.6	6.4	7.7
LESA [22]	ZSL	19.4	31.6	28.7
	GZSL	5.6	14.4	16.8
BiAM [35]	ZSL	25.8	32.0	29.4
	GZSL	8.9	15.5	18.5
Our Approach	ZSL	**29.4**	**32.8**	**32.3**
	GZSL	**10.2**	**15.8**	**19.2**

Table 2. State-of-the-art comparison for **multi-label ZSL and GZSL** tasks on the **Open-Images**-V4 dataset. We show the indicators of F1-Score in the case of $K \in 10, 20$ and mAP. Best results are shown in bold.

Method	**Task**	**mAP**	**F1 (K = 10)**	**F1 (K = 20)**
CONSE [36]	ZSL	40.4	0.4	0.3
	GZSL	43.5	2.6	2.4
LabelEM [2]	ZSL	40.5	0.5	0.4
	GZSL	45.2	5.2	5.1
Fast0Tag [49]	ZSL	41.2	0.7	0.6
	GZSL	45.2	16.0	13.0
Attention per Cluster [22]	ZSL	40.7	1.2	0.9
	GZSL	44.9	16.9	13.5
LESA [22]	ZSL	41.7	1.4	1.0
	GZSL	45.4	17.4	14.3
BiAM [35]	ZSL	62.8	4.1	3.7
	GZSL	79.6	17.6	15.1
Our Approach	ZSL	**65.7**	**7.5**	**6.5**
	GZSL	**79.9**	**27.6**	**24.1**

Figure 5 shows the mAP, inference time, and GFLOPs comparisons between our model for obtaining semantic information based on channel responses and the two methods (*LESA* [22] and *BiAM* [35]) for acquiring semantic information based on spatial features and achieving optimal results. In the mAP comparison, it can be seen that we have the highest accuracy for prediction in the ZSL task. At the same time, due to the small amount of data to be processed, the inference speed is the fastest of all comparison methods when we use the same GPU of NVIDIA RTX 3090. Finally, precisely because the model only needs to deal with a single-dimensional channel response, our $(\mathrm{ML})^2$P-Encoder module requires much less computation than *LESA* and *BiAM* that deal with spatial attention. At the same time, the feature map is grouped to avoid the geometric increase of the computational complexity caused by the feature pyramid. This shows that our $(\mathrm{ML})^2$P-Encoder can be more efficient.

Table 3. **Ablation study** shows the contribution of the different components in our proposed approach. The baseline methods are performed on the *NUS-WIDE* test-set.

		a	b	c	d	ours
Forward Pyramid			✓	✓	✓	✓
$(\mathrm{ML})^2$P-Encoder				✓		✓
Group Attention					✓	✓
mAP	ZSL	25.3	27.3	28.4	27.9	**29.4**
	GZSL	8.1	8.5	9.2	8.8	**10.2**

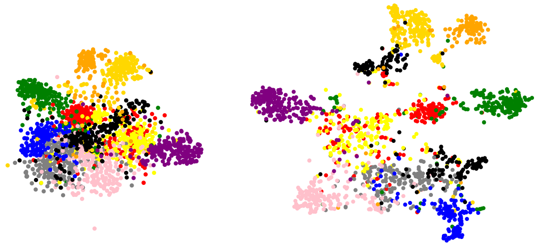
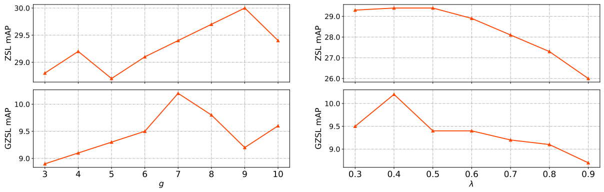

(a) W/O $(\mathrm{ML})^2$P-Encoder (b) With $(\mathrm{ML})^2$P-Encoder

Figure 3. **Evaluation of t-SNE** (zoom in for a better view).

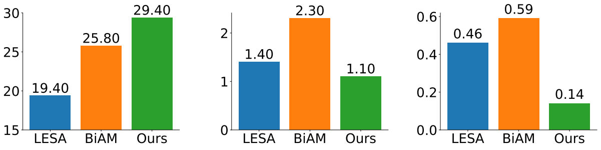
4.3. Hyper-parameter Selection

Our method includes two hyper-parameters, the number of groups g and the weight of the regularization term λ. We use the control variable method. In terms of initializing hyper-parameters, the number of output semantic vectors g is set to 7, and the value of λ is set to 0.4. The line graph in Figure 4 shows the mAP results achieved on the ZSL and GZSL tasks with different hyper-parameters, respectively. In addition, we can also see the impact of changes in hyper-parameters on the prediction accuracy of the model. It can be seen that the number of g does not have a very significant effect on the mAP of the ZSL task. But the impact on GZSL is more obvious. After comparison, we believe that when $g = 7$, two different tasks can be well balanced. For the choice of the value of λ, we found that its change will have a greater impact on mAP. But only when $\lambda = 0.4$, the performance of GZSL is far better than other results, and ZSL also achieves the optimal result. So the optimal hyper-parameters we choose $g = 7$ and $\lambda = 0.4$.



(a) g (b) Weights

Figure 4. **Hyper-Parameter selection.**The higher the mAP the better. All the experiments are performed on the *NUS-WIDE* test-set.



(a) mAP (b) Inference time (ms) (c) GFLOPs

Figure 5. **Comparison of our $(\mathrm{ML})^2$P-Encoder with BiAM and LESA in mAP, inference time, and FLOPs.**The higher the mAP the better, the lower the Inference time and GFLOPs the better. All methods are performed on the *NUS-WIDE* test-set.

4.4. Ablation Study

Ablation Study: To illustrate the effectiveness of each module designed in our method, we arrange three comparative experiments. The specific results are shown in Table 3. As the most primitive structure, model 'a' only contains shuffle and grouping operations. But after adding the 'Forward Pyramid', the model expands the number of features. As the number of optional feature channels increases, the amount of information brought by the channel also increases, thus achieving more competitive results. The addition of $(\mathrm{ML})^2$P-Encoder enables the model to process the channel response of specific classes. The supplement of Group Attention is to give the model-specific information for solving multi-label tasks, that is, inter-class correlation. The combination of $(\mathrm{ML})^2$P-Encoder and Group Attention greatly improves the prediction ability of the model in ZSL and GZSL tasks, indicating that our model construction has achieved great success.

t-SNE: Figure 3 shows the performance of $(\mathrm{ML})^2$P-Encoder in t-SNE visualization. It can be seen that after using $(\mathrm{ML})^2$P-Encoder, the boundaries of inter-class become much clearer, proving the correctness of our exploration for class-specific channel responses.

Different Backbones: Table 4 shows the results produced by our method using different backbones. It can be seen from the results that ResNet [20] has obvious advantages over VGG [37]. As the ResNet network deepens and the number of parameters increases, the results obtained by our model become better. This is exactly in line with the result variation of an end-to-end model.

Table 4. Our **C^3-MLZSL** approach with **different backbones** for multi-label ZSL and GZSL tasks on the **NUS-WIDE** dataset. We show the indicators of F1-Score in the case of $K \in 3, 5$ and mAP. The best results are shown in bold.

Backbones	**Task**	**mAP**	**F1 (K = 3)**	**F1 (K = 5)**
VGG19 [37]	ZSL	29.4	32.8	32.3
	GZSL	10.2	15.8	19.2
ResNet50 [20]	ZSL	30.9	33.6	33.2
	GZSL	10.7	15.9	19.4
ResNet101 [20]	ZSL	31.2	33.9	33.9
	GZSL	10.9	16.1	19.5

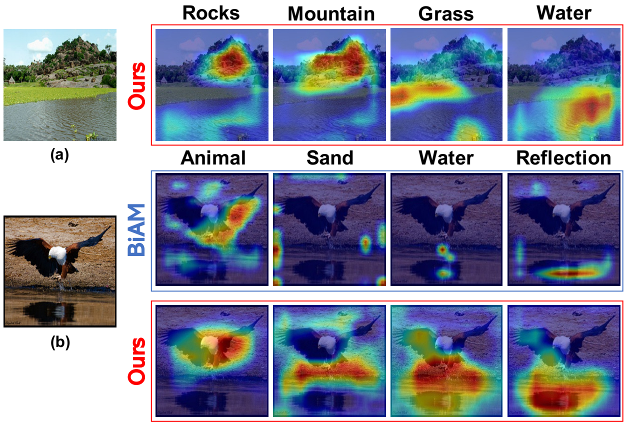


Figure 6. **Attention visualization.** where (a) is the attention response of our **C^3-MLZSL** when faced with unseen labels. (b) is the comparison of attention visualization results of our **C^3-MLZSL** and **BiAM** [35] models. See appendix for more results.

4.5. Multi-Label Learning

Table 5 shows the results of the model for multi-label image classification. The baselines we compare include not only state-of-the-art MLZSL models, but also multi-label image classification models including *Logistic Regression* [40], *WSABIE* [43], *WARP* [17] and *CNN-RNN* [42]. As can be seen from the results, our model far surpasses many multi-label image classification models and the classic *Fast0Tag* [49] algorithm in mAP performance. This is because the above models only process the input image into a single semantic vector, and limited image embedding cannot build the semantic diversity for multi-label classification. For other methods such as *LESA* [22] and *BiAM* [35], they noticed that the attention regions of different objects in multi-label images are different, and thus define the label-related embeddings from the perspective of the spatial domain. However, after feature extraction, our model takes into account that the channel response can be important information representing the class semantics, and this superior performance just verifies the rationality of the exploration.

Table 5. Performance of **Multi-label image classification** task on *NUS-WIDE* datasets. The best results are in bold.

Method	**F1(K=3)(↑)**	**F1(K=5)(↑)**	**mAP(↑)**
Logistic [40]	51.1	46.1	21.6
WARP [17]	54.4	49.4	3.1
WSABIE [43]	53.8	49.2	3.1
Fast0Tag [49]	53.8	48.6	22.4
CNN-RNN [42]	55.2	50.8	28.3
Kim *et al.* [23]	56.8	51.3	32.6
LESA ApC [22]	56.6	50.7	31.7
LESA [22]	58.0	52.0	31.5
BiAM [35]	59.6	53.4	47.8
Ours	**59.8**	**53.8**	**48.0**

5. Conclusion

In this paper, we focus on the neglect of channel-wise class information and over-reliance on spatial-wise class information in previous MLZSL models, then propose C^3-MLZSL structure and the (ML)^{2}P-Encoder component. The C^3-MLZSL structure first group multi-scale features, then use the (ML)^{2}P-Encoder to calculate the correlation of channels within each group and perform information fusion to get the semantic vectors. These semantic vectors are then aggregated through group attention to learn mutual information between groups. Finally, the model successfully learns channel-class correlation. Extensive experiments on the large-scale *NUS-WIDE* and *Open-Images*-V4 datasets show that our model has achieved very competitive results on MLZSL compared with other state-of-the-art models.

6. Acknowledgment

This research was supported by fundings from the Key-Area Research and Development Program of Guangdong Province (No. 2021B0101400003), Hong Kong RGC Research Impact Fund (No. R5060-19), Areas of Excellence Scheme (AoE/E-601/22-R), General Research Fund (No. 152203/20E, 152244/21E, 152169/22E, 152211/23E), Shenzhen Science and Technology Innovation Commission (JCYJ20200109142008673), the National Natural Science Foundation of China (No. 62102327), and PolyU Internal Fund (No. P0043932).

References

[1] Zeynep Akata, Mateusz Malinowski, Mario Fritz, and Bernt Schiele. Multi-cue zero-shot learning with strong supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 59–68, 2016. 2

[2] Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. Label-embedding for image classification. *IEEE transactions on pattern analysis and machine intelligence*, 38(7):1425–1438, 2015. 5, 6

[3] Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. Label-embedding for image classification. *IEEE transactions on pattern analysis and machine intelligence*, 38(7):1425–1438, 2016. 3

[4] Maxime Bucher, Stéphane Herbin, and Frédéric Jurie. Improving semantic embedding consistency by metric learning for zero-shot classiffication. In *European Conference on Computer Vision*, pages 730–746. Springer, 2016. 3

[5] Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning*, pages 89–96, 2005. 5

[6] Soravit Changpinyo, Wei-Lun Chao, Boqing Gong, and Fei Sha. Synthesized classifiers for zero-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5327–5336, 2016. 3

[7] Zhi Chen, Yadan Luo, Sen Wang, Ruihong Qiu, Jingjing Li, and Zi Huang. Mitigating generation shifts for generalized zero-shot learning. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 844–852, 2021. 3

[8] Zhao-Min Chen, Xiu-Shen Wei, Peng Wang, and Yanwen Guo. Multi-label image recognition with graph convolutional networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5177–5186, 2019. 2

[9] Xing Cheng, Hezheng Lin, Xiangyu Wu, Fan Yang, Dong Shen, Zhongyuan Wang, Nian Shi, and Honglin Liu. Mltr: Multi-label classification with transformer. *arXiv preprint arXiv:2106.06195*, 2021. 1

[10] Tat-Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, and Yantao Zheng. Nus-wide: a real-world web image database from national university of singapore. In *Proceedings of the ACM international conference on image and video retrieval*, pages 1–9, 2009. 5

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 5

[12] Shay Deutsch, Soheil Kolouri, Kyungnam Kim, Yuri Owechko, and Stefano Soatto. Zero shot learning via multi-scale manifold regularization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7112–7119, 2017. 2

[13] Thibaut Durand, Nazanin Mehrasa, and Greg Mori. Learning a deep convnet for multi-label classification with partial labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 647–657, 2019. 1, 2

[14] Lei Feng, Bo An, and Shuo He. Collaboration based multi-label learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3550–3557, 2019. 1, 2

[15] Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. *Advances in neural information processing systems*, 26, 2013. 2, 3

[16] Bin-Bin Gao and Hong-Yu Zhou. Learning to discover multi-class attentional regions for multi-label image recognition. *IEEE Transactions on Image Processing*, 30:5920–5932, 2021. 1, 2

[17] Yunchao Gong, Yangqing Jia, Thomas Leung, Alexander Toshev, and Sergey Ioffe. Deep convolutional ranking for multilabel image annotation. *arXiv preprint arXiv:1312.4894*, 2013. 1, 2, 8

[18] Omkar Gune, Biplab Banerjee, Subhasis Chaudhuri, and Fabio Cuzzolin. Generalized zero-shot learning using generated proxy unseen samples and entropy separation. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 4262–4270, 2020. 3

[19] Jingcai Guo and Song Guo. A novel perspective to zero-shot learning: Towards an alignment of manifold structures via semantic feature expansion. *IEEE Transactions on Multimedia*, 23:524–537, 2020. 2

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1, 7, 8

[21] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016. 4

[22] Dat Huynh and Ehsan Elhamifar. A shared multi-attention framework for multi-label zero-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8776–8786, 2020. 2, 3, 5, 6, 8

[23] Jin-Hwa Kim, Jaehyun Jun, and Byoung-Tak Zhang. Bilinear attention networks. *arXiv preprint arXiv:1805.07932*, 2018. 5, 6, 8

[24] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[25] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016. 2

[26] Elyor Kodirov, Tao Xiang, and Shaogang Gong. Semantic autoencoder for zero-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3174–3183, 2017. 2, 3

[27] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 951–958. IEEE, 2009. 3

[28] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Attribute-based classification for zero-shot visual object categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(3):453–465, 2014. 3

[29] Chung-Wei Lee, Wei Fang, Chih-Kuan Yeh, and Yu-Chiang Frank Wang. Multi-label zero-shot learning with structured knowledge graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1576–1585, 2018. 3

[30] Jingjing Li, Mengmeng Jing, Lei Zhu, Zhengming Ding, Ke Lu, and Yang Yang. Learning modality-invariant latent representations for generalized zero-shot learning. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1348–1356, 2020. 3

[31] Yanan Li, Donghui Wang, Huanhang Hu, Yuetan Lin, and Yueting Zhuang. Zero-shot recognition using dual visual-semantic mapping paths. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3279–3287, 2017. 2

[32] Teng Long, Xing Xu, Youyou Li, Fumin Shen, Jingkuan Song, and Heng Tao Shen. Pseudo transfer with marginalized corrupted attribute for zero-shot learning. In *Proceedings of the 26th ACM international conference on Multimedia*, pages 1802–1810, 2018. 3

[33] Thomas Mensink, Efstratios Gavves, and Cees GM Snoek. Costa: Co-occurrence statistics for zero-shot classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2441–2448, 2014. 2

[34] Pedro Morgado and Nuno Vasconcelos. Semantically consistent regularization for zero-shot recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6060–6069, 2017. 2

[35] Sanath Narayan, Akshita Gupta, Salman Khan, Fahad Shahbaz Khan, Ling Shao, and Mubarak Shah. Discriminative region-based multi-label zero-shot learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8731–8740, 2021. 2, 3, 5, 6, 8

[36] Mohammad Norouzi, Tomas Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg S Corrado, and Jeffrey Dean. Zero-shot learning by convex combination of semantic embeddings. In *2nd International Conference on Learning Representations, ICLR 2014*, 2014. 2, 3, 5, 6

[37] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1, 4, 5, 6, 8

[38] Richard Socher, Milind Ganjoo, Christopher D Manning, and Andrew Ng. Zero-shot learning through cross-modal transfer. In *Advances in neural information processing systems*, pages 935–943, 2013. 3

[39] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, Andrew Rabinovich, et al. Going deeper with convolutions. Cvpr, 2015. 1

[40] Grigorios Tsoumakas and Ioannis Katakis. Multi-label classification: An overview. *International Journal of Data Warehousing and Mining (IJDWM)*, 3(3):1–13, 2007. 8

[41] Andreas Veit, Neil Alldrin, Gal Chechik, Ivan Krasin, Abhinav Gupta, and Serge Belongie. Learning from noisy large-scale datasets with minimal supervision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 839–847, 2017. 5

[42] Jiang Wang, Yi Yang, Junhua Mao, Zhiheng Huang, Chang Huang, and Wei Xu. Cnn-rnn: A unified framework for multi-label image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2285–2294, 2016. 1, 2, 8

[43] Jason Weston, Samy Bengio, and Nicolas Usunier. Wsabie: Scaling up to large vocabulary image annotation. In *Twenty-Second International Joint Conference on Artificial Intelligence*, 2011. 2, 8

[44] Yongqin Xian, Zeynep Akata, Gaurav Sharma, Quynh Nguyen, Matthias Hein, and Bernt Schiele. Latent embeddings for zero-shot classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 69–77, 2016. 3

[45] Yongqin Xian, Bernt Schiele, and Zeynep Akata. Zero-shot learning-the good, the bad and the ugly. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4582–4591, 2017. 2, 3

[46] Hsiang-Fu Yu, Prateek Jain, Purushottam Kar, and Inderjit Dhillon. Large-scale multi-label learning with missing labels. In *International conference on machine learning*, pages 593–601. PMLR, 2014. 1, 2

[47] Chenrui Zhang, Xiaoqing Lyu, and Zhi Tang. Tgg: Transferable graph generation for zero-shot and few-shot learning. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 1641–1649, 2019. 3

[48] Li Zhang, Tao Xiang, and Shaogang Gong. Learning a deep embedding model for zero-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2021–2030, 2017. 2

[49] Yang Zhang, Boqing Gong, and Mubarak Shah. Fast zero-shot image tagging. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5985–5994. IEEE, 2016. 2, 3, 5, 6, 8

[50] Ziming Zhang and Venkatesh Saligrama. Zero-shot learning via joint latent similarity embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6034–6042, 2016. 3

[51] Yizhe Zhu, Mohamed Elhoseiny, Bingchen Liu, Xi Peng, and Ahmed Elgammal. A generative adversarial approach for zero-shot learning from noisy texts. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 3