Deep0Tag: Deep Multiple Instance Learning for Zero-shot Image Tagging

Shafin Rahman, *Student Member, IEEE,* Salman Khan and Nick Barnes, *Member, IEEE*

***Abstract*—Zero-shot learning aims to perform visual reasoning about unseen objects. In-line with the success of deep learning on object recognition problems, several end-to-end deep models for zero-shot recognition have been proposed in the literature. These models are successful in predicting a single unseen label given an input image but do not scale to cases where multiple unseen objects are present. Here, we focus on the challenging problem of *'zero-shot image tagging'*, where multiple labels are assigned to an image, that may relate to objects, attributes, actions, events, and scene type. Discovery of these scene concepts requires the ability to process multi-scale information. To encompass global as well as local image details, we propose an automatic approach to locate relevant image patches and model image tagging within the Multiple Instance Learning (MIL) framework. To the best of our knowledge, we propose the first end-to-end trainable deep MIL framework for the multi-label zero-shot tagging problem. We explore several alternatives for instance-level evidence aggregation and perform an extensive ablation study to identify the optimal pooling strategy. Due to its novel design, the proposed framework has several interesting features: (1) Unlike previous deep MIL models, it does not use any off-line procedure (e.g., Selective Search or EdgeBoxes) for bag generation. (2) During test time, it can process any number of unseen labels given their semantic embedding vectors. (3) Using only image-level seen labels as weak annotation, it can produce a localized bounding box for each predicted label. We experiment with the large-scale NUS-WIDE and MS-COCO datasets and achieve superior performance across conventional, zero-shot and generalized zero-shot tagging tasks.**

***Index Terms*—Deep learning, Multiple instance learning, Feature pooling, Object detection, Zero-shot tagging**

I. Introduction

Due to the advancement of multimedia technology, a significantly large volume of multimedia data has become available to us. For example, enormous growth in online photo collections requires automatic image tagging algorithms that can provide both seen and unseen labels to the images. Thus, image tagging and tag-based retrieval has emerged as a principal direction of multimedia research [1]. Since the number of possible (query) tags is infinite, we need to adopt techniques that can assign image tags beyond the limited tag set available during training. Zero-shot learning (ZSL) is the obvious solution to this problem as it seeks to assign unseen tags during inference. Few notable works adopting ZSL on multimedia applications are person re-identification [2], video retrieval [3] and image representation [4].

In recent years, most of the zero-shot classification methods assign only a single unseen tag/category label to an image [5]–[11]. However in real-life settings, multimedia images often come with multiple objects or concepts that may or may not be observed during training. Despite the importance and practical nature of this problem, there are very few existing methods with the capability to address the zero-shot image tagging task [12]–[15]. This is primarily due to the challenging nature of the problem. In this paper, we identify three pertinent issues that underpin the zero-shot image tagging task. *First*, any object or concept can either be present at a localized region or be inferred from the holistic scene information (e.g., *'sun'* vs *'sunset'*). *Second*, objects and concepts are often occluded in natural scenes and scene context provides valuable information for image tagging. *Third*, the assignment of multiple image tags (seen and unseen) requires an accurate mapping function from visual to semantic domain. Moreover, the available label space is significantly larger comprising of thousands of possible tags, and an ideal zero-shot framework should have the flexibility to incorporate new unseen tags on the fly during testing.

To address the above-mentioned challenges, we introduce the first unified network for zero-shot image tagging termed '*Deep0Tag*', that automatically locates relevant image patches, learns their discriminative representations and assigns relevant tags in a single framework. We note that the previous approaches [14], [16], [18]–[21] used off-the-shelf features or applied off-line procedures like Selective Search [23], EdgeBoxes [24] or BING [25] for patch generation which served as a bag-of-instances. The reliance on the external non-differentiable procedure in either feature extraction or bag generation means they cannot be trained end-to-end for the image tagging task. Our solution is based on three key novelties that systematically tackle the aforementioned requirements for zero-shot image tagging. These include,

(a) Multi-scale Concept Discovery: We propose an automatic procedure to extract useful patches at multiple scales that collectively form a bag of visual instances (Sec. III-B1). A distinguishing feature of our approach is that the bag-of-instances not only encodes the global information about a scene but also has a rich representation for localized object-level features. Note that the existing attempts on zero-shot tagging (e.g., [12]–[15]) mostly used image-level global features. These techniques, therefore, work only for the most prominent objects but often fail for non-salient concepts due to the lack of localized information.

S. Rahman is with the Research School of Engineering, The Australian National University, Canberra, ACT 2601, Australia, and also with Data61, Commonwealth Scientific and Industrial Research Organization, Canberra, ACT 2601, Australia (e-mail: shafin.rahman@anu.edu.au).

S. Khan is with the Research School of Engineering, The Australian National University, Canberra, ACT 2601, Australia, and also with the Inception Institute of Artificial Intelligence, Abu Dhabi 00000, UAE.

N. Barnes is with the Research School of Engineering, The Australian National University, Canberra, ACT 2601, Australia, and also with Data61, Commonwealth Scientific and Industrial Research Organization, Canberra, ACT 2601, Australia.

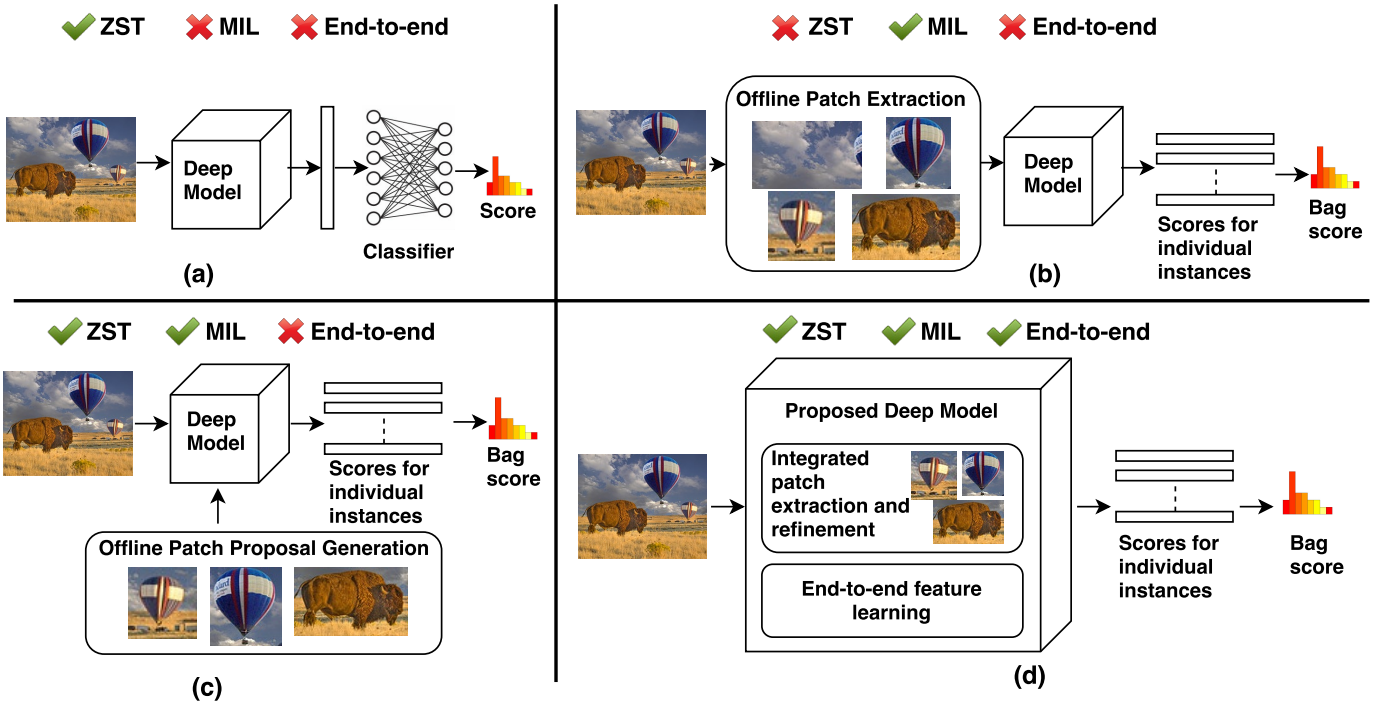

Fig. 1: Overview of different multi-label image annotation architectures. (a) [14], [16], [17] extract deep features separately, then feed those features to a neural network for classification. (b) [18], [19] use an external procedure to extract image patches for bag generation then feed those patches to a deep model to get final bag score for seen classes. (c) [20]–[22] process whole image as well as patches obtained using an external process to get bag scores for seen classes. (d) Our proposed MIL model simply takes an image as an input and produces bag score for both seen and unseen classes.

(b) Context Aggregation: Deep0Tag introduces a Multiple Instance Learning (MIL) framework that fuses multi-scale information encoded in the bag generated from each image (Sec. III-B2). MIL assumes that each bag contains at least one instance of the true labels defined by the ground-truth. We introduce two schemes to aggregate contextual information in an image i.e., semantic and visual domain fusion. Further, we show that both parametric and non-parametric pooling techniques can be used with our proposed MIL framework that can effectively model general functions computed on a bag (Corollary 1,2).

(c) Incorporating Semantics: Our proposed network maps local and global information in a bag to a semantic embedding space such that the correspondences with both seen and unseen classes can be found. We use a margin maximizing loss formulation in the semantic domain that specifically focuses on hard cases during training (Sec. III-B5). The closest to our work are [14], [21], that also employ a visual-semantic mapping to bridge the gap between seen and unseen classes. However, these models are not fully learnable due to pre-trained features or off-line bag extraction.

In Figure 1, we illustrate a block diagram to compare different kinds of image tagging frameworks proposed in the literature. Our framework integrates the concept discovery, context aggregation and semantics incorporation in a single framework that can be jointly trained in an end-to-end manner. Therefore, it encompasses different modules of MIL into only a single integrated network, which results in state-of-the-art performance for conventional and zero-shot image tagging. The proposed framework leads to several advantages for image tagging: (1) It can work in both conventional and zero-shot settings, (2) It is extendable to any number of novel tags from an open vocabulary as it does not use any prior information about the unseen concepts, (3) The proposed method can function in a weakly-supervised setting, i.e., it can annotate a bounding box for both seen and unseen tags without using any ground-truth bounding box during training.

A preliminary version of this work appeared in [26] where the contribution was restricted to mean and max-pooling during semantic domain aggregation. The current extended version includes: (1) a more detailed study on information aggregation in both semantic and visual domains, (2) the introduction of both fixed (mean, max and LSE based) and learnable pooling (attention based) mechanisms for aggregation, (3) detailed theoretical motivation for loss formulation, and (4) a substantial number of new experimental evaluation and ablation analysis on aggregation mechanism. The rest of the paper is organized as follows: Sec. II provides a brief introduction to related work. Sec. III details the proposed tagging framework. Our results with comparisons and ablation analysis are provided in Sec. IV and the paper concludes in Sec. V.

II. Related Work

1) Multiple instance learning (MIL): MIL in combination with deep neural networks has been used for multi-label classification [17], [19], tagging [21], image captioning [27], text analysis [28] and medical imaging [29]. In most of these cases, MIL depends on max or mean pooling. However, the log-sum-exp pooling is designed to generalize traditional mean/max pooling [30], [31]. Lack of trainable parameters is the apparent drawback of those pooling strategies. To address this, both linearly [28] and non-linearly [32] trainable pooling mechanisms have been proposed in recent years. In this paper, we have experimented with an end-to-end learnable deep MIL framework employing various kinds of pooling strategies for zero-shot image tagging task. Another related feature pooling strategy is proposed in [33]. Different from our work, they do not consider learned pooling mechanisms, decision level pooling and semantic information as they work in a non-ZSL setup.

2) Zero-shot learning: In recent years, we have seen exciting progress on Zero Shot Learning (ZSL). The overall goal of ZSL approaches is to classify an image to an unseen class for which no training is performed. Investigations are focused on domain adaptation [8], class attribute association [34], unsupervised semantics [8], hubness effect [7], generalized ZSL setting [5], [35] of inductive [6] or transductive ZSL learning [9] and zero-shot object detection [36]–[38]. The major shortcoming of most of these approaches is their inability to assign multiple labels to an image, which is a major limitation in real-world settings. In line with the general consideration of only a single label per image, traditional ZSL methods use recognition datasets which mostly contain only one prominent concept per image. Here, we present an end-to-end deep zero shot tagging method that can assign multiple tags per image.

3) End-to-end object detection: Image tags can correspond to either whole image or a specific location within it [39]. To model the relationship between a tag and its corresponding locations, we intend to localize objects in a scene. To do so, we are interested in end-to-end object detection frameworks. Popular examples of such frameworks are Faster R-CNN [40], R-FCN [41], SSD [42] and YOLO [43]. The main distinction among these models is the object localization process. R-CNN [40] and R-FCN [41] used a Region Proposal Network (RPN) to generate object proposals whereas SSD [42] and YOLO [43] propose bounding box and classify it in a single step. The latter group of models usually run faster, but they are relatively less accurate than the first group. Some recent approaches attempt to improve the performance of both single and double stage detectors by proposing a feature pyramid network and a loss function to handle foreground-background imbalance [44], [45]. Due to the focus on highly accurate object detection, we built on Faster R-CNN [40] as the backbone architecture in the current work.

4) Zero-shot image tagging: Instead of assigning one unseen label to an image during a recognition task, zero-shot tagging allows assigning multiple relevant unseen tags to an image and/or a ranking for unseen tags array. Although interesting, this problem is not well-addressed in the zero-shot learning literature. An early attempt at this, extended a zero-shot recognition approach [46] to perform zero-shot tagging by proposing a hierarchical semantic embedding to make the label embedding more reliable [15]. [13] proposed a transductive multi-label version of the problem where a predefined and relatively small set of unseen tags were considered. In a recent work, [14] proposed a fast zero-shot tagging approach that can be trained using only seen tags and tested using both seen and unseen tags. In another recent work, [47] framed the tagging task as a multi-label zero-shot learning problem and proposed a structured knowledge graph to propagate inter-dependencies among seen and unseen classes. [21] proposed a multi-instance visual-semantic embedding approach that can extract localized image features. The main drawback of these early efforts is their dependence on pre-trained CNN features (in [13]–[15], [47]) or fast-RCNN [22] features (in [21]) and therefore not end-to-end trainable. Moreover, the reliance on pre-trained networks violates the zero-shot protocol since some of the unseen categories overlap with the classes used to perform pre-training of the deep network. In this work, we propose a fully end-to-end solution for both conventional and zero-shot tagging.

III. Our Method

A. Problem Formulation

Suppose, we have a set of 'seen' tags denoted by $\mathcal{S} = \{1, \ldots, \mathrm{S}\}$ and another set of 'unseen' tags $\mathcal{U} = \{\mathrm{S}+1, \ldots, \mathrm{S}+\mathrm{U}\}$, such that $\mathcal{S} \cap \mathcal{U} = \phi$ where, S and U represents the total number of seen and unseen tags respectively. We also denote $C = \mathcal{S} \cup \mathcal{U}$, such that $\mathrm{C} = \mathrm{S} + \mathrm{U}$ is the cardinality of the tag-label space. For each of the tags $c \in C$, we can obtain a 'd' dimensional word vector $\mathbf{v}_c$ (word2vec or GloVe) as a semantic embedding. The training examples can be defined as a set of tuples, $\{(\mathbf{X}_s, \mathbf{y}_s) : s \in [1, M]\}$, where $\mathbf{X}_s$ is the s^{th} input image and $\mathbf{y}_s \subset \mathcal{S}$ is the set of relevant seen tags. We represent u^{th} testing image as $\mathbf{X}_u$ which corresponds to a relevant seen and/or unseen tag $\mathbf{y}_u \subset C$. Note that, $\mathbf{X}_u$, $\mathbf{y}_u$, $\mathcal{U}$ and its corresponding word vectors are not observed during training. Now, we define the following problems:

- *Conventional tagging:* Given $\mathbf{X}_u$ as input, assign relevant seen tags $\mathbf{y}_u \subset \mathcal{S}$.
- *Zero-shot tagging (ZST):* Given $\mathbf{X}_u$, assign relevant unseen tags $\mathbf{y}_u \subset \mathcal{U}$.
- *Generalized zero-shot tagging (GZST):* Given $\mathbf{X}_u$ as input, assign relevant tags from both seen and unseen $\mathbf{y}_u \subset C$.

MIL formulation: We formulate the above mentioned problem definitions in Multiple Instance Learning (MIL) framework. Let us represent the s^{th} training image with a bag of $n+1$ instances $\mathbf{X}_s = \{\mathbf{x}_{s,0} \ldots \mathbf{x}_{s,n}\}$, where i^{th} instance $\mathbf{x}_{si}$ represents either an image patch (for $i > 0$) or the complete image itself (for $i = 0$). We assume that each instance $\mathbf{x}_{s,i}$ has an individual label $\ell_{s,i}$ which is unknown during training. As $\mathbf{y}_s$ represents relevant seen tags of $\mathbf{X}_s$, according to the MIL assumption, the bag has at least one instance for each tag in

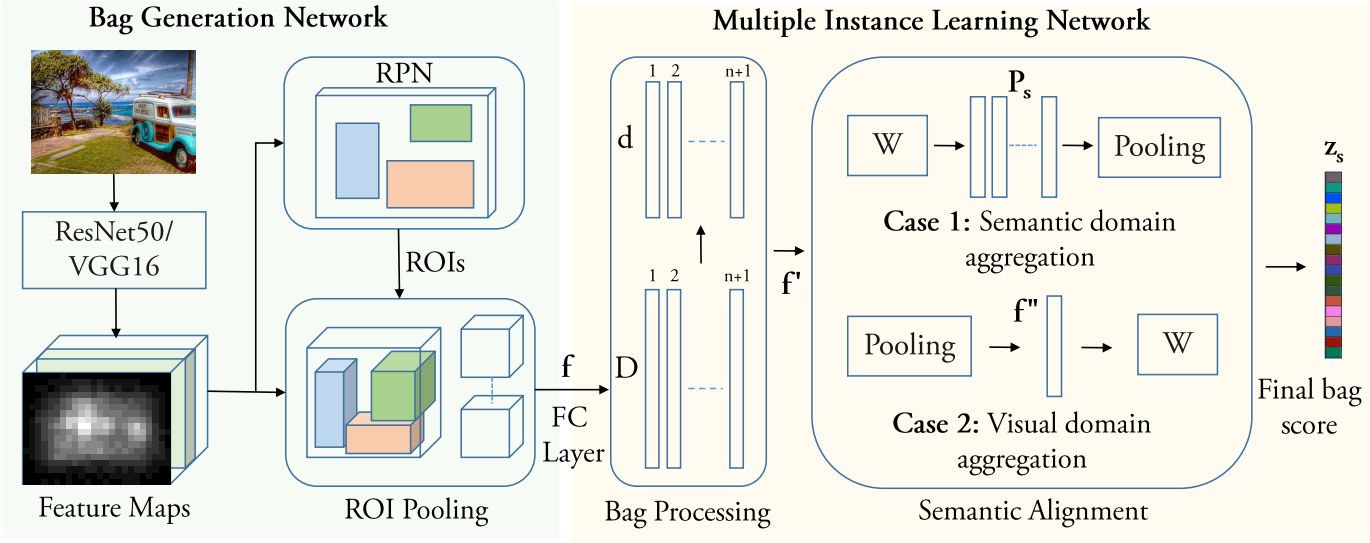

Fig. 2: Proposed *Deep0Tag* architecture for MIL based zero-shot image tagging.

the set $\mathbf{y}_s$ and no instance for $\{\mathcal{S} \setminus \mathbf{y}_s\}$ tags. The bag- and instance-level labels are related by:

$$y \in \mathbf{y}_s \quad \text{iff} \quad \sum_{i=0}^{n} [\![\ell_{s,i} = y]\!] > 0, \quad \forall y \in \mathcal{S}. \tag{1}$$

Thus, instances in $\mathbf{X}_s$ can work as a positive example for $y \in \mathbf{y}_s$ and negative example for $y' \in \{\mathcal{S} \setminus \mathbf{y}_s\}$. This formulation does not use instance level tag annotation which makes it a weakly supervised problem. Our aim is to design and learn an end-to-end deep learning model that can itself generate the appropriate bag-of-instances and simultaneously assign relevant tags to the bag.

B. Network Architecture

The proposed network architecture is illustrated in Fig. 2. It is composed of two parts: bag generation network (*left*) and Multiple Instance Learning (MIL) network (*right*). The bag generation network generates a bag-of-instances as well as their visual features, and the MIL network processes the resulting bag of instance features to find the final multi-label prediction which is calculated by a global pooling of the prediction scores of individual instances. In this manner, bag generation and zero-shot prediction steps are combined in a unified framework that effectively transfers learning from seen to unseen tags.

1) Bag generation: In our proposed method, the bag contains image patches which are assumed to cover all objects and concepts presented inside the image. Many closely related traditional methods [18]–[21] apply some external procedure such as Selective Search [23], EdgeBoxes [24] or BING [25] for this purpose. Such a strategy creates three problems: **(1)** the off-line external process does not allow an end-to-end learnable framework, **(2)** the patch generation process is prone to more frequent errors because it can not be fine-tuned on the target dataset, and **(3)** the MIL framework needs to process patches rather than the image itself. In this paper, we propose to solve these problems by generating a useful bag of patches by the network itself. The recent achievements of object detection frameworks, such as the Faster-RCNN [40], allow us to generate object proposals and later perform detection within a single network. We adopt this strategy to generate a bag of image patches for MIL. Remarkably, the original Faster-RCNN model is designed for supervised learning, while our MIL framework extends it to weakly supervised setting.

A Faster-RCNN model [40] with Region Proposal Network (RPN) is learned using the ILSVRC-2017 detection dataset. This architecture uses a base network ResNet-50 [48] which is shared between RPN and classification/localization network. As practiced, the base network is initialized with pre-trained weights. Though not investigated in this paper, other popular CNN models, e.g., VGG [49] and GoogLeNet [50] can also be used as the shared base network. Now, given a training image $\mathbf{X}_s$, the RPN can produce a fixed number (n) of region of interest (ROI) proposals $\{\mathbf{x}_{s,1} \dots \mathbf{x}_{s,n}\}$ with a high recall rate. For image tagging, all tags may not represent an object. Rather, tags can be concepts that describe the whole image, e.g., nature and landscape. To address this issue, we add a global image ROI (denoted by $\mathbf{x}_{s,0}$) comprising of the complete image to the ROI proposal set generated by the RPN. Afterwards, ROIs are fed to ROI-Pooling and subsequent densely connected layers to calculate D-dimensional features set: $\mathcal{F}_s = [\mathbf{f}_{s,0} \dots \mathbf{f}_{s,n}] \in \mathbb{R}^{D \times (n+1)}$ where $\mathbf{f}_{s,0}$ is the feature representation of the whole image. This bag is then forwarded to MIL network for prediction.

2) MIL Network: Our network design then comprises of two component blocks within the MIL network: '*bag processing block*' and '*semantic alignment block*'. The bag processing block has two fully connected layers with a non-linear activation ReLU. The role of this block is to remap the bag of features to the dimension of semantic embedding space by calculating $\mathcal{F}'_s = [\mathbf{f}'_{s,0} \dots \mathbf{f}'_{s,n}] \in \mathbb{R}^{d \times (n+1)}$. $\mathcal{F}'_s$ is forwarded to the semantic alignment block. This block performs two important operations to calculate the final score for each bag:

(i) semantic projection and (ii) MIL pooling operation. Based on the sequence of this two operations this block can be implemented in two ways.

Case 1- Semantic domain aggregation: In this case, the visual domain features are first mapped to semantic space and their responses are aggregated. Specifically, given a bag of instance features $\mathcal{F'}_s$, we first compute the prediction scores of individual instances, $\mathbf{P}_s = [\mathbf{p}_{s,0} \dots \mathbf{p}_{s,n}] \in \mathbb{R}^{\mathrm{S}\times(n+1)}$ by projecting them onto the fixed semantic embedding, $\mathbf{W} = [\mathbf{v}_1 \dots \mathbf{v}_\mathrm{S}] \in \mathbb{R}^{d\times\mathrm{S}}$, containing word vectors of seen tags,

$$\mathbf{P}_s = \mathbf{W}^T \mathcal{F'}_s. \tag{2}$$

Since the supervision is only available for bag-level predictions (i.e., image tags), we require an aggregation mechanism $\mathcal{A}(\cdot)$ to combine predictions scores $\mathbf{P}_s$ for individual instances in a bag. Using the semantic-domain aggregation, we can obtain the final bag score as follows:

$$\mathbf{z}_s = \mathcal{A}\Big(\{\mathbf{p}_{s,0}, \mathbf{p}_{s,1}, \dots, \mathbf{p}_{s,n}\}\Big). \tag{3}$$

Case 2- Visual domain aggregation: In this case, visual features are first aggregated using a pooling operation and then transformed to semantic domain. Specifically, given a bag of instance features $\mathcal{F'}_s$, we perform a pooling operation first to obtain a universal feature representation of bag $\mathbf{f}'_s$:

$$\mathbf{f}''_s = \mathcal{A}\Big(\{\mathbf{f}'_{s,0}, \mathbf{f}'_{s,1} \dots \mathbf{f}'_{s,n}\}\Big). \tag{4}$$

After that, we project $\mathbf{f}''_s$ onto the semantic embedding space to calculate the final score of the bag:

$$\mathbf{z}_s = \mathbf{W}^T \mathbf{f}''_s. \tag{5}$$

3) Aggregation Mechanism: The aggregation mechanism mentioned in Eqs. 3 and 4 can be implemented using a non-parametric (fixed) or a parametric (learnable) function. In both categories, we explore a range of pooling methods as described below.

Fixed Pooling: Given a set of input feature vectors $\{\mathbf{i}_j\}_{j=0}^n$, the aggregated output $\mathbf{o}$ can be obtained via max, mean or log-sum-exp (LSE) pooling as follows:

$$\mathbf{o} = \max\Big\{\mathbf{i}_0, \mathbf{i}_1, \dots, \mathbf{i}_n\Big\}, \tag{6}$$

$$\mathbf{o} = \frac{1}{n+1}\sum_{j=0}^{n} \mathbf{i}_j, \tag{7}$$

$$\mathbf{o} = \frac{1}{r}\log\Big[\frac{1}{n+1}\sum_{j=0}^{n}\exp(r\mathbf{i}_j)\Big], \tag{8}$$

where r is a fixed hyper-parameter during network training. In our case, the input vectors are either $\mathbf{p}_{s,j}$ or $\mathbf{f}'_{s,j}$ for case 1 and 2 respectively.

Learned Pooling: The mean, max or log-sum-exp based pooling approaches described above do not have any trainable parameters. To address this issue, we experiment with an attention based pooling strategy recently proposed by Ilse *et al.* [32]. Given a set of input features, the attention based pooling operation can be summarized as follows:

$$\mathbf{o} = \sum_{j=0}^{n} a_j \mathbf{i}_j \quad \text{where, } a_j = \frac{\exp\{\mathbf{u}^T \tanh(\mathbf{V}\mathbf{i}_j)\}}{\sum_{k=0}^{n}\exp\{\mathbf{u}^T \tanh(\mathbf{V}\mathbf{i}_j)\}} \tag{9}$$

where, $\mathbf{V} \in \mathbb{R}^{L\times d}$ and $\mathbf{u} \in \mathbb{R}^{L\times 1}$ are learnable parameters which are a part of the pooling operation. As tanh(.) is approximately linear, [32] also proposed the following gated attention mechanism to increase non-linearly:

$$a_j = \frac{\exp\{\mathbf{u}^T \tanh(\mathbf{V}\mathbf{i}_j) \odot \mathrm{sigmoid}(\mathbf{V'}\mathbf{i}_j)\}}{\sum_{k=0}^{n}\exp\{\mathbf{u}^T \tanh(\mathbf{V}\mathbf{i}_k) \odot \mathrm{sigmoid}(\mathbf{V'}\mathbf{i}_k)\}} \tag{10}$$

where, $\mathbf{V'} \in \mathbb{R}^{L\times d}$ denotes learnable parameters and $\odot$ represents element-wise multiplication. The learned pooling strategies are not suitable for semantic domain aggregation because the parameter dimension then becomes dependent on number of seen tags, S. As the number of seen and unseen tags may not be same, the network cannot predict unseen scores during testing. For visual domain aggregation case, the dimension of learnable pooling parameters $\mathbf{V}$ or $\mathbf{V'}$ is dependent on the bag feature dimension which remains fixed during training and testing. Therefore, we only investigate learned pooling for the visual domain aggregation.

4) Theoretical Analysis: It can be proved that the proposed MIL framework can approximate any general function defined on the bag $\mathbf{X}_s$. Since our formulation is based on object proposals $\mathbf{x}_{s,i}$, we can approximate a general function $\mathcal{G}(\cdot)$ on the bag with the following object level decomposition:

$$\mathcal{G}(\mathbf{X}_s = \{\mathbf{x}_{s,i}\}_0^n) \approx h(\{g(\mathbf{x}_{s,i})\}_0^n),\ s.t.,\ g(\mathbf{x}_{s,i}) = \begin{cases}\mathbf{p}_{s,i}, & \text{case 1}\\ \mathbf{f}'_{s,i}, & \text{case 2}\end{cases} \tag{11}$$

where $g(\cdot)$ is the transformation function defined using a deep network and $h(\cdot)$ is a symmetric function that is invariant to permutations of the object proposals. Such a decomposition is intuitive because the proposal set is unordered and its cardinality can vary, neither of these two factors should effect the bag level predictions. The function $\mathcal{G}(\cdot)$ can be approximated adequately using the symmetric transformations in Eq. 6, Eq. 7 or Eq. 8 according to the following corollaries:

Corollary 1: Max Pooling – From the Theorem of Universal approximation for continuous set functions [51], a Hausdorff symmetric function $\mathcal{G}(\cdot)$ can be approximated with in the bounds $\epsilon \in \mathbb{R}^+$ if $g(\cdot)$ is a continuous function and $h(\cdot)$ is a element-wise vector maximum operator (denoted as 'max'), i.e.,

$$\parallel \mathcal{G}(\mathbf{X}_s) - \max\{g(\mathbf{x}_{s,i})\} \parallel < \epsilon. \tag{12}$$

Corollary 2: Mean Pooling – From the the Chevalley-Shephard-Todd (CST) theorem [52], [53], a permutation invariant continuous function $\mathcal{G}(\cdot)$ operating on the set $\mathbf{X}_s$ can be arbitrarily approximated if $g(\cdot)$ is a transformation function implemented as a neural network, where neural networks are universal approximators [54], and $h(\cdot)$ is an element-wise mean operator (denoted as 'mean'), i.e.,

$$\mathcal{G}(\mathbf{X}_s) \approx \mathrm{mean}\{g(\mathbf{x}_{s,i})\}. \tag{13}$$

Remark: A smoother version of the above functions called Log-Sum-Exp (LSE) (Eq. 8) is often followed for convex approximation [30]. For example, it is helpful to approximate a non-differentiable function like max operation.

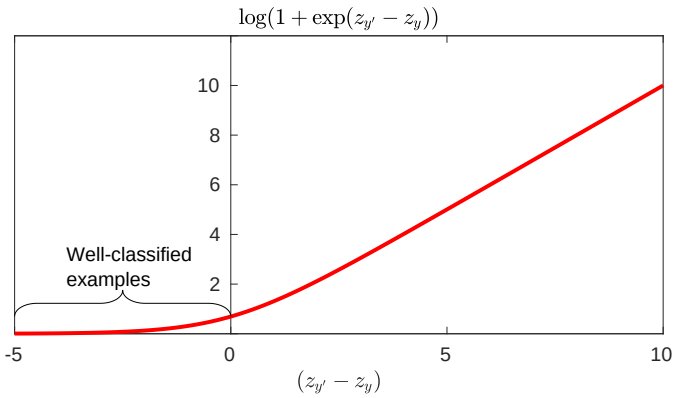

Fig. 3: The red curve shows the loss function used in this work that focuses more on its mistakes and directly maximizes the gap between positive and negative predictions.

$$\mathcal{G}(\mathbf{X}_s) \approx \frac{1}{r}\log\Big[\frac{1}{n+1}\sum_{i=0}^{n}\exp(rg(\mathbf{x}_{s,i}))\Big], \tag{14}$$

where, $|\mathcal{G}(\mathbf{X}_s) - \max\{g(\mathbf{x}_{s,i})\}| < \frac{n+1}{r}$. Here, r controls the amount of smoothness where the high and low value for r tend to behave like the max and mean function respectively [31]. Therefore, our MIL formulation can learn a permutation invariant function on bags of visual instances consisting of both object and concept representations.

5) Loss formulation: Suppose, for s^{th} training image, $\mathbf{z}_s = [z_1 \dots z_S]$ contains final multi-label prediction of a bag for seen classes. This bag is a positive example for each tag $y \in \mathbf{y}_s$ and negative examples for each tag $y' \in \{\mathcal{S} \setminus \mathbf{y}_s\}$. Thus, for each pair of y and y', the difference $z_{y'} - z_y$ represents the disparity between predictions for positive and negative tags. Our goal is to minimize these differences in each iteration. We formalize the loss of a bag considering it to contain both positive and negative examples for different tags:

$$L_{tag}(\mathbf{z}_s, \mathbf{y}_s) = \frac{1}{|\mathbf{y}_s||\mathcal{S} \setminus \mathbf{y}_s|}\sum_{y' \in \{\mathcal{S}\setminus\mathbf{y}_s\}}\sum_{y \in \mathbf{y}_s}\log\Big(1 + \exp(z_{y'} - z_y)\Big).$$

We minimize the overall loss on all training images as follows:

$$L = \arg\min_{\Theta}\frac{1}{M}\sum_{s=1}^{M}\Big(L_{tag}(\mathbf{z}_s, \mathbf{y}_s)\Big).$$

Here, Θ denote the parameter set of the proposed network and M is the total number of training images.

We argue that the above loss formulation is better suited for multi-label classification as compared to the standard cross-entropy loss. This can be explained from the loss curve in Fig. 3. For well classified cases, scores for positive tags (z_y) are higher than negative ones ($z_{y'}$), i.e., $z_{y'} - z_y < 0$ which gives a small loss penalty after log function is applied. It works opposite in other cases, which brings the following benefits: **(1)** It provides an inherent mechanism to tackle data imbalance by assigning much less penalty to an already correct prediction compared to the incorrect ones. This criteria helps to heavily focus on incorrect predictions during training and speeds up the overall learning process, **(2)** Our loss can impose a ranking penalty alongside forcing to predict a specific ground truth. This capability is especially important to deal with word vectors because image features need to align with a high dimensional vector, not just a specific value. We note that a similar rank based penalty was proposed in [55]. However, different from our case, the formulation in [55] focuses mainly on the largest error term among all the differences $z_{y'} - z_y$. In contrast, our formulation aligns all features to their corresponding word vectors by penalizing *each* difference based on the quality of the alignment.

6) Prediction: During testing, we modify the fixed embedding $\mathbf{W}$ to include seen and unseen word vectors instead of only seen word vectors. Suppose, after modification $\mathbf{W}$ becomes $\mathbf{W}' = [\mathbf{v}_1 \dots \mathbf{v}_S, \mathbf{v}_{S+1} \dots \mathbf{v}_{S+U}] \in \mathbb{R}^{d\times C}$. With the use of $\mathbf{W}'$ in Eq. 2 and 5 for cases 1 and 2 respectively, we get prediction scores of both seen and unseen tags for each individual instance in the bag. Then, after the global pooling step, we get the final prediction score for each seen and unseen tags. Finally, based on the tagging task (conventional/zero-shot/generalized zero-shot), we assign top K target tags (from the set $\mathcal{S}$, $\mathcal{U}$ or C) with higher scores to an input image.

IV. Experiments

A. Setup

1) Dataset: We perform our experiments using a real-world web image dataset namely NUS-WIDE [56]. It contains 269,648 images with three sets of image tags from Flickr. The first, second and third set contains 81, 1000 and 5018 tags respectively. The tags inside the first set are carefully chosen, therefore less noisy whereas the third set has the highest noise in annotations. Following the previous work [14], we use 81 tags from the first set as unseen in this paper. We notice that the tag 'interesting' comes twice within the second set. After removing this inconsistency and selecting 81 unseen tags from the second set results in 924 tags which we use as seen for our experiments. The dataset provides the split of 161,789 training and 107,859 testing images. We use this recommended setting while ignoring the untagged images.

2) Visual and semantic embedding: Unlike previous attempts on zero-shot tagging [14], [15], our model works in an end-to-end manner using ResNet-50 [48] and VGG16 [49] as a base network. It means the visual feature are originating from ResNet-50/VGG16, but they are updated during iterations. As the semantic embedding, we use ℓ_2 normalized 300 dimensional GloVe vectors [57]. We are unable to use word2vec embedding [58] because the pre-trained word-vector model cannot provide vectors for all of the 1005 (924 seen + 81 unseen) tags.

3) Evaluation metric: Following the work [14], we calculate precision (P), recall (R) and F-1 score (F1) of the top K predicted tags ($K = 3$ and 5 is used) and Mean image Average Precision (MiAP) as evaluation metrics. The following equation is used to calculate MiAP of an input image I:

$$MiAP(I) = \frac{1}{|\mathcal{R}|}\sum_{j=1}^{|\mathcal{T}|}\frac{q_j}{j}\delta(I, t_j),$$

where, $|\mathcal{R}|$ = total number of relevant tags, $|\mathcal{T}|$ = total number of ground truth tags, q_j = number of relevant tags of j^{th} rank and $\delta(I, t_j) = 1$ if j^{th} tag t_j is associated with the input image I, otherwise $\delta(I, t_j) = 0$.

4) Training details: The Faster-RCNN model is first pre-trained on ILSVRC-2017 object detection dataset. After that, the last two layers (i.e. the classification and localization head) are removed to produce bag generation network. We used the following settings during Faster-RCNN [40] training: rescaling shorter size of image as 600 pixels, RPN stride = 16, three anchor box scale 128, 256 and 512 pixels, three aspect ratios 1:1, 1:2 and 2:1, non-maximum suppression (NMS) with IoU threshold = 0.7 with maximum number of proposals = 300. The network predicts scores for S number of seen tags by fine-tuning on target tagging dataset i.e. NUS-WIDE [56]. During the training of our MIL framework, we generated one bag-of-instances at each iteration from an image to feed our network. We chose a fixed n number of RoIs proposed by the RPN which archives the best objectness score. We carried out 774k training iterations using Adam optimizer with a learning rate of 10^{-5}, $\beta_1 = 0.9$ and $\beta_2 = 0.999$. We implemented our model in *Keras* library.

B. Tagging Performance

In this subsection, we evaluate the performance of our framework on three variants of tagging as introduced in Sec. III-A, namely conventional, zero-shot and generalized zero-shot tagging. The results of these three tasks are summarized next. Notably, we first compare our best performing model based on visual domain pooling (case 2 with LSE pooling and bag size 32) with other top-performing methods in Sec. IV-B1, IV-B2, IV-B3 and later provide a detailed ablation study exploring different variants of our proposed model in Sec. IV-B4.

1) Compared methods: To compare our results, we have reimplemented two closely related published methods (ConSE [46] and Fast0Tag [14]) and one simple baseline based on ResNet-50 CNN architecture. We choose these methods for comparison because of their suitability to perform zero-shot tasks. We provide a brief introduction to compared methods below:

- **ConSE [46]:** It was originally introduced for zero-shot learning for image classification. This approach first learns a classifier for seen tags and generates a semantic embedding for unseen input by linearly combining word vectors of seen classes using seen prediction scores. In this way, it can rank unseen tags based on the distance of generated semantic embedding and the embedding of unseen tags.
- **Fast0Tag [14]:** This method is the main competitor of our work. It is a deep feature-based approach, where features are calculated from a pre-trained VGG-19 [49]. Afterward, a neural network is trained on these features to classify seen and unseen input. This approach outperforms many established methods like WRAP [59], WSABIE [60], TagProp [61], FastTag [62] on conventional tagging task. Therefore, in this paper, we do not consider those low-performing methods for comparison. The performance reported in this paper using Fast0Tag method is relatively different from the published results because of few reasons: (1) We use the ResNet-50/VGG16 whereas [14] reported results on VGG-19, (2) Although [14] experimented on NUS-WIDE, they only used a subset of 223,821 images in total, (3) The implementation for [14] did not consider the repetition of the seen tag 'interesting'.
- **Baseline:** The baseline method is a special case of our proposed method which uses the whole image as a single instance inside the bag. It breaks the multiple instance learning consideration but does not affect the end-to-end nature of the proposed solution.

2) Results: We conduct experiments on the conventional, zero-shot and generalized zero-shot settings and report results in Tables I and II. In all of our experiments, the same test images are used. The basic difference between conventional vs. zero-shot tagging is whether the 81 tags set is used during training or not. For the zero-shot settings, we perform our training with 924 seen tags and test on 81 unseen tags. However, in the conventional tagging case, all tags are considered as seen and training+testing is performed on the same tag set. For the generalized zero-shot tagging case, the same testing image set is used, but instead of predicting tags from 81 tag set, our method predicts tags from 1005 tag set (924 seen, 81 unseen).

We compare with two state-of-the-art methods in Tables I and II. Our method outperforms other methods by a significant margin. Notably, the following observations can be developed from the results: **(1)** The performance of conventional tagging is much better than zero-shot case because unseen tags and associated images are not present during training for zero-shot tasks. One can consider that the performance of conventional case is an upper-bound for zero-shot tagging case. **(2)** Similar to previous work [5], the performance for the generalized zero-shot tagging task is even poorer than the zero-shot tagging task. This can be explained by the fact that the network gets biased towards the seen tags and scores low on unseen categories. This subsequently leads to a decrease in performance. **(3)** Similar to the observation from [14], Fast0Tag beats ConSE [46] in zero-shot cases. The main reason is that the ConSE [46] does not use semantic word vectors during its training which is crucial to find a bridge between seen and unseen tags. No results are reported with ConSE for the conventional tagging case because it is only designed for zero-shot scenarios. **(4)** The baseline beats other two compared methods in most of the cases (except VGG16 Fast0tag vs. Baseline) because of the end-to-end training while incorporating word vectors in the learning phase. This approach is benefited by the appropriate adaptation of feature representations for the tagging task. **(5)** The performances of ConSE, Fast0Tag, and Baseline (where local features are not used) are better with VGG16 than ResNet50. It tells us that VGG16 works better as a global feature extractor compared to ResNet50. **(6)** Our approach outperforms all other competitors because it utilizes localized image features based on MIL, performs end-to-end

Method	Network	MiAP	$K = 3$			$K = 5$		
			P	R	F1	P	R	F1
Fast0Tag [14]	ResNet50	35.73	20.24	34.48	25.51	16.16	45.87	23.90
Baseline	ResNet50	40.45	22.95	39.09	28.92	17.99	51.09	26.61
Ours	ResNet50	52.50	33.77	57.53	42.55	22.21	63.06	32.85
Fast0Tag [14]	VGG16	50.43	32.21	54.88	40.59	21.25	60.33	31.43
Baseline	VGG16	49.82	32.21	54.87	40.59	21.24	60.31	31.42
Ours	VGG16	**53.56**	**34.43**	**58.66**	**43.39**	**22.55**	**64.02**	**33.35**

TABLE I: Results for conventional tagging. K denotes the number of assigned tags.

Method	Zero-shot tagging							Generalized zero-shot tagging						
	MiAP	$K = 3$			$K = 5$			MiAP	$K = 3$			$K = 5$		
		P	R	F1	P	R	F1		P	R	F1	P	R	F1
ResNet50														
ConSE [46]	18.91	8.39	14.30	10.58	7.16	20.33	10.59	7.27	2.11	3.59	2.65	8.82	5.69	6.92
Fast0Tag [14]	24.73	13.21	22.51	16.65	11.00	31.23	16.27	10.36	5.21	8.88	6.57	12.41	8.00	9.73
Baseline	29.75	16.64	28.34	20.97	13.49	38.32	19.96	12.07	5.99	10.20	7.54	14.28	9.21	11.20
Ours	**39.21**	**25.69**	**43.77**	**32.38**	**17.22**	**48.89**	**25.46**	**20.41**	**33.78**	**13.07**	**18.85**	**23.65**	**15.25**	**18.54**
VGG16														
ConSE [46]	32.30	20.39	34.74	25.70	13.86	39.35	20.50	12.89	22.47	8.70	12.54	15.50	9.99	12.15
Fast0Tag [14]	35.55	23.22	39.55	29.26	15.61	44.32	23.09	18.26	30.18	11.68	16.84	21.09	13.60	16.54
Baseline	35.19	23.06	39.29	29.07	15.67	44.49	23.17	18.33	29.99	11.61	16.74	21.13	13.63	16.57
Ours	37.25	24.32	41.44	30.66	16.35	46.43	24.18	18.87	30.90	11.96	17.24	21.61	13.94	16.95

TABLE II: Results for zero-shot and generalized zero-shot tagging tasks.

Method	Network	MiAP	K=3			K=5		
			P	R	F1	P	R	F1
ConSE [46]	ResNet50	0.36	0.08	0.06	0.07	0.10	0.13	0.11
Fast0Tag [14]	ResNet50	3.26	3.15	2.40	2.72	2.51	3.18	2.81
Baseline	ResNet50	3.61	3.51	2.67	3.04	2.83	3.59	3.16
Ours	ResNet50	**6.61**	**6.52**	**4.96**	**5.63**	**5.25**	**6.66**	**5.87**
ConSE [46]	VGG16	0.47	0.12	0.09	0.11	0.14	0.18	0.16
Fast0Tag [14]	VGG16	5.19	5.10	3.88	4.40	4.11	5.21	4.59
Baseline	VGG16	5.09	5.00	3.81	4.32	4.08	5.18	4.57
Ours	VGG16	5.98	5.85	4.45	5.05	4.67	5.93	5.23

TABLE III: Results for zero-shot tagging task with 4,084 unseen tags.

training and integrates semantic vectors of seen tags within the network. We also illustrate some qualitative comparisons in Fig. 5.

3) Tagging in the wild: Since our method does not use any information about unseen tags in zero-shot settings, it can process an infinite number of unseen tags from an open vocabulary. We test with such a setting using the 5018 tag set of NUS-WIDE. We remove 924 seen tags and ten other tags for which no GloVe vectors were found (handsewn, interestingness, manganite, marruecos, mixs, monochromia, shopwindow, skys, topv and uncropped) to produce a large set of 4084 unseen tags. After training with 924 seen tags, the performance of zero-shot tagging with this set is shown in Table III. Because of the extreme noise in these annotations, the results are very poor in general, but our method still outperforms other competitors by a reasonable margin [14], [46].

4) Ablation study: As mentioned earlier, our proposed architecture can be implemented in two different ways: semantic domain aggregation (case 1) or visual domain aggregation (case 2). For both of the cases, different kinds of pooling mechanisms (like global mean, max, log-sum-exp (LSE), attention and gated attention) could be employed. In Table IV, we perform an extensive ablation study on the ZST and GZST tasks with the different combinations of network architectures. We also include two non-MIL-instance based approaches (`Ins-mean` and `Ins-max`) where features are treated as an individual instance rather than a part of the bag. In this particular case, no pooling is required, and loss is calculated based on the mean/max of all instances loss. Some of the key findings of this ablation analysis are as follows: **(1)** `Ins-max` performs worse than `Ins-mean` because `Ins-max` calculates loss based on only one object proposal ignoring the contribution of all other possible proposals. **(2)** The case when visual domain aggregation (case 2) is done results in better performance than the semantic domain aggregation (case 1). In case 2, the pooling fuses visual features that combine global and local details to find a single overall representation of the scene. In the projection step, this overall representation is projected onto the word vectors. In contrast, case 1 projects the global and local features to the word vectors directly. The projection of case 2 aligns the features to word embeddings better than that of case 1 because localized features are extracted from noisy object proposals (as trained to detect objectness) and the pooling on feature level can suppress that noise before the alignment. **(3)** LSE outperforms mean or max pooling in both cases because LSE has a combined effect of

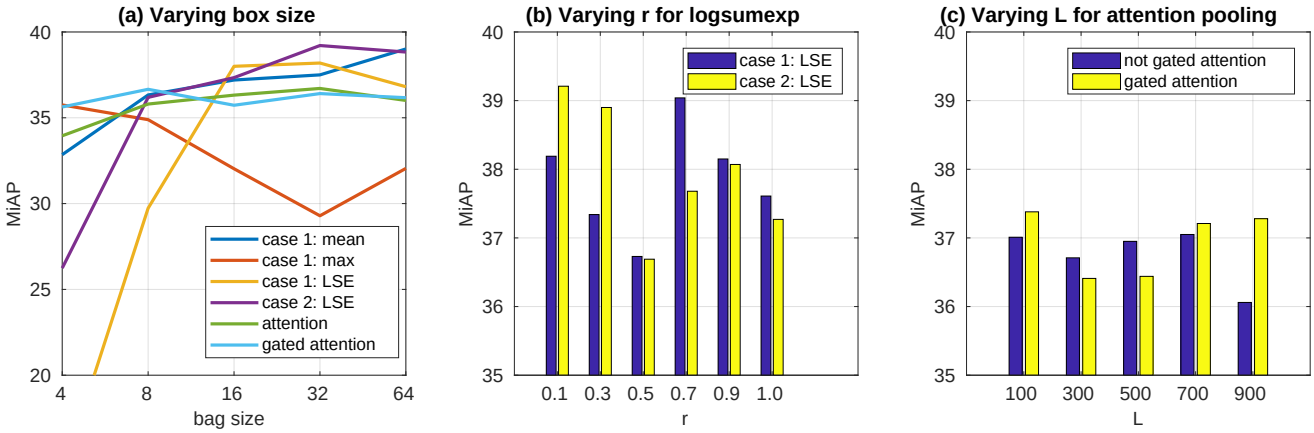

Fig. 4: Ablation study: impact of (a) pooling type and bag size (b) hyper-parameter r on LSE pooling (c) size L on attention based pooling while performing zero-shot tagging task using ResNet50 backbone.

Pooling type	ZST							GZST							ZST in wild						
	MiAP	K=3			K=5			MiAP	K=3			K=5			MiAP	K=3			K=5		
		P	R	F1	P	R	F1		P	R	F1	P	R	F1		P	R	F1	P	R	F1
Ins-mean	37.7	24.9	42.4	31.3	16.7	47.4	24.7	20.8	33.5	12.9	18.7	23.4	15.1	18.3	6.4	6.3	4.8	5.4	5.0	6.4	5.7
Ins-max	13.9	10.2	17.4	12.8	6.50	18.5	9.6	5.3	8.7	3.4	4.9	6.5	4.2	5.1	0.6	0.1	0.1	0.1	0.1	0.1	0.1
Case1-mean	37.5	21.2	36.1	26.7	16.6	47.2	24.6	20.5	27.7	10.7	15.4	23.7	15.3	18.6	5.8	5.4	4.1	4.7	4.4	5.6	4.9
Case1-max	21.5	16.9	28.8	21.3	12.4	35.3	18.4	11.1	17.8	6.9	10.0	13.7	8.9	10.8	1.2	1.0	0.7	0.8	0.8	1.0	0.8
Case1-LSE	38.2	25.1	42.8	31.7	16.8	47.8	24.9	**21.4**	**35.2**	**13.6**	**19.7**	**24.6**	**15.8**	**19.3**	6.5	6.4	4.9	5.5	5.1	6.5	5.7
Case2-mean	38.2	25.1	42.8	31.6	16.8	47.6	24.8	20.3	33.6	13.0	18.7	23.8	15.1	18.3	5.9	5.7	4.3	4.9	4.6	5.8	5.1
Case2-max	36.0	23.9	40.7	30.1	16.2	45.9	23.9	17.9	30.1	11.6	16.8	21.1	13.6	16.5	5.3	5.1	3.9	4.4	4.2	5.3	4.7
Case2-LSE*	**39.2**	**25.7**	**43.8**	**32.4**	**17.2**	**48.9**	**25.5**	20.4	33.8	13.1	18.8	23.7	15.3	18.5	**6.6**	**6.5**	**5.0**	**5.6**	**5.3**	**6.7**	**5.9**
att	36.7	24.3	41.5	30.7	16.4	46.5	24.2	19.0	31.4	12.2	17.5	22.3	14.2	17.3	5.9	5.8	4.4	5.0	4.6	5.8	5.2
att-gated	36.4	23.9	40.7	30.1	16.1	45.7	23.8	19.0	31.6	12.2	17.7	22.2	14.3	17.4	5.5	5.3	4.0	4.5	4.3	5.5	4.8

TABLE IV: Ablation study: different pooling strategies using bag size 32 and ResNet50 backbone architecture. Hyper-parameters $r = 0.1$ and $L = 300$ are used in LSE and attention related experiments respectively. The overall best model is marked with *.

both pooling types. **(4)** Although the attention based pooling (gated and not-gated) has trainable pooling parameters, still it cannot outperform LSE based pooling. Again, the reason is the noisy nature of objectness bounding boxes which confuses the learning of pooling parameters.

5) Insights on Pooling: Pooling after semantic projection (case 1) works on prediction scores, which means it is a decision-level fusion scheme. It summarizes the semantic projection of individual features in a bag by an overall projection score of all features. Thus, it forces each individual region in an image to correctly align itself with its true word-vectors. This approach can be understood as a bottom-up approach where an overall prediction is made based on the individual region-based predictions.

In contrast, pooling before semantic projection (case 2) works on image features. It summarizes the feature representation of all possible locations into one global feature representation of the image. Then, the global representation is projected onto the semantic space only once to align it with the word vectors of true classes. This approach can be considered as a top-down approach which uses global information to make an overall prediction about image tags.

Our experiments show that the feature level fusion (case 2) results in better performance. This is due to the fact that the fused visual features incorporate wide context available in a scene which better models inter-tag relationships and holistic scene information. We also note that case 1 is more useful when tags relate to local details as this strategy can better locate individual instances of objects.

It is also important to note that the projection in case 2 aligns the features to word embeddings better than that of case 1. This is because the localized features are extracted from noisy object proposals (as trained to detect objectness) and the pooling on feature level can suppress that noise before the alignment. In fact, Case 2 is useful when the image contains more stuff (non-object) categories or abstract concepts because in that case, global representation can pick the abstract concepts and/or interrelation between them to tag an image.

Since LSE pooling outperforms other pooling strategies in most cases, we are particularly interested in its behavior when the value of its hyper-parameter 'r' is changed. In Figure 4(b), we vary the value of r from 0.1 to 1.0 which transitions its behavior from mean pooling to max pooling. We observe that a low value of r works better in case 2 and the opposite is true for case 1. The variation can be attributed to the difference between the visual (case 2) and semantic space (case 1) features. Similarly, in Figure 4(c), we vary the size of L of attention based pooling. As gated attention has strong non-linearity, it outperforms the non-gated version most of the times.

6) Analysis on Bag Size: The proposed frameworks can work for different numbers of instances in the bag. In Figure 4(a), we perform an ablation study for zero-shot tagging based on different combinations of network settings. The optimal bag size depends on the dataset and pooling type. We notice that a large bag-size generally degrades tagging performance for max pooling and vice-versa for all other pooling types. This variation is related to the noise inside the tag annotation of the ground truth. Many previous deep MIL networks [19], [20] recommended max-pooling for MIL where they experimented on object detection dataset containing the ground-truth annotation without any label noise. In contrast, other than the 81-tag set, NUS-WIDE contains significant noise in the tag annotations. Therefore, LSE and mean-pooling with large bag size achieve a balance in the noisy tags, outperforming max-pooling in general for NUS-WIDE. Notably, the bag size of our framework is far smaller compared to other MIL approaches [19], [20]. We observe that with only a small numbers of instances (e.g., 4) in the bag, our method can beat state-of-the-art approaches [14], [46]. Being dependent on external bag generator [23]–[25] previous methods lose control inside the generation process. Thus, a large bag size helps them to get enough proposals to choose the best score in last max-pooling layer. Conversely, our method controls the bag generation network by fine-tuning shared ResNet-50/VGG16 layers which eventually can relax the requirement of large bag sizes.

7) Cross-entropy vs. Our proposed loss: In Table V, we compare our loss with traditional multi-class cross-entropy (CE) loss. As stated earlier, CE loss tries to predict a specific ground-truth without considering inter-class differences that results in a poor alignment among visual features and word vectors. In contrast, our proposed loss tries to increase between class differences by assigning ranking penalty (high and low for incorrect and correct classification respectively). Therefore, our proposed loss outperforms CE by a significant margin.

8) Effect of reverse mapping: During bag processing, the MIL network remaps $D-$dimensional visual feature $\mathcal{F}_s \in \mathbb{R}^{D\times(n+1)}$ to $\mathcal{F}_s \in \mathbb{R}^{d\times(n+1)}$ in order to match the dimension of semantic word vectors. Because of this remapping, our semantic alignment strategies (Cases 1 and 2) work on semantic space. However, instead of mapping visual features to semantic space, a reverse mapping semantic to visual could be performed [7]. In that case, Eq. 2 and 5 will change to:

$$\mathbf{P}_s = \mathbf{W}^T U_1 \mathcal{F}_s \text{ and } \mathbf{z}_s = \mathbf{W}^T U_2 \mathbf{f}_s,$$

where, $\mathbf{f}_s = \mathcal{A}\Big(\{\mathbf{f}_{s,0}, \mathbf{f}_{s,1} \dots \mathbf{f}_{s,n}\}\Big)$, U_1 and $U_2 \in \mathbb{R}^{\mathrm{S}\times D}$ are learnable parameters to map $\mathbf{W} \in \mathbb{R}^{d\times \mathrm{S}}$ to $\mathbf{W}^T U \in \mathbb{R}^{D\times \mathrm{S}}$ (D dimensional visual feature domain). In Table V, we compare both visual to semantic and semantic to visual strategies. We achieve slightly better performance with visual to semantic approach. This trend is opposite to the proposal of established zero-shot learning approach [7]. One possible reason is that we learn the features along with word vectors whereas [7] used fixed pre-trained visual features and only train word vectors. Another reason is that our ranking based loss is different from the traditional ℓ_2 loss used in [7].

Setup	MiAP	$K = 3$			$K = 5$		
		P	R	F1	P	R	F1
CE	5.79	2.10	3.57	2.64	1.39	3.95	2.06
S $\rightarrow$ V	37.17	24.42	41.60	30.77	16.49	46.82	24.39
V $\rightarrow$ S	39.21	25.69	43.77	32.38	17.22	48.89	25.46
Baseline (187)	29.85	19.89	33.88	25.06	13.59	38.59	20.10
Ours (187)	38.45	25.16	42.87	31.71	16.95	48.12	25.07

TABLE V: Comparison between different setups of our approach. CE: Multi-class cross-entropy loss on our `Case1-LSE` architecture. S $\rightarrow$ V: Semantic to visual domain projection, V $\rightarrow$ S: Visual to semantic domain projection, Baseline (187) and our (187): performance using the pre-trained models with 187 non-overlapping classes.

9) Effect of pre-training: In this paper, we use a Faster RCNN model pre-trained on ILSVRC-DET 2017 dataset to initialize the bag generation network. We have identified 13 ILSVRC-DET 2017 classes (bear, birds, cars, cat, dog, fox, horses, person, plane, tiger, train, whales, zebra) which are also present within 81 unseen classes of NUS-WIDE dataset. One can argue that these common unseen classes can benefit from the pre-trained RPN. As we train the whole model in the end-to-end manner after excluding these common classes, we note that the advantage for these unseen classes is negligible. In Table V, we report performance after initializing the model with the pre-trained weights that were learned on 187 classes (without considering the 13 overlapping classes) and notice similar performance as for the case when all 200 classes in ILSVRC-DET 2017 dataset are used. This is due to the fact that RPN is trained (during Faster RCNN training) in a class agnostic fashion. Thus, regardless of seen or unseen tags presented in an image, it can always detect bounding boxes based on objectness measure. As a result, the exclusion of overlapping classes does not make a significant difference in performance.

C. Multi-label classification on MS-COCO

We experiment with the large scale MSCOCO-2014 dataset [63] and tackle both conventional and zero-shot tagging problems. This dataset has 80 object classes with 82,783 and 40,504 training and validation images, respectively. Being a tagging task, we ignore all bounding box annotation during training. For all experiments, we again use 300-dimensional GloVe vectors as a semantic embedding and ResNet50 as a backbone architecture.

1) Conventional tagging: This experiment aims at the multi-label classification problem where all tags strictly represent only MSCOCO classes. Following the experiment settings of Lee *et al.* [47], we use 40,137 validation images after removing without-label images to test on this task. In Table VI (a), we compare our approach with several other established methods in the literature. Our approach consistently outperforms others because it considers multi-scale information comprising of both local and global cues, and due to the end-to-end nature of the solution. We notice WARP [59] and Fast0Tag [14] achieved identical performance (also reported in Lee *et al.* [47]) because both methods used a similar loss (taking a single global feature as input) that applies a ranking

(a) Multi-label classification results

Method	P	R	F1
WSABIE [60]	59.3	61.3	60.3
WARP [59]	60.2	62.2	61.2
Fast0Tag [14]	60.2	62.2	61.2
Lee *et al.* [47]	**74.1**	64.5	69.0
Ours	69.9	**72.2**	**71.1**

(b) Zero-shot and generalized zero-shot tagging results

Method	ZST			GZST		
	P	R	F1	P	R	F1
ConSE [46]	11.4	28.3	16.2	23.8	28.8	26.1
Fast0Tag [14]	24.7	61.4	35.3	38.5	46.5	42.1
Baseline	23.7	59.0	33.9	35.1	42.4	38.4
Our (bag=64)	**26.5**	**65.9**	**37.8**	**43.2**	**52.2**	**47.3**

TABLE VI: Experiments on on MS-COCO [63] with K=3.

penalty by maximizing the prediction difference of positive and negative tags. Therefore, in conventional settings, both methods behave similarly. Note that WARP does not have zero-shot learning capability. In contrast, although Fast0Tag can work for a conventional setting, it is specially designed for the zero-shot learning task.

2) Zero-shot experiments: This task requires splitting MSCOCO classes into seen and unseen sets. Recently Bansal *et al.* [37] proposed a split of 48 seen and 17 unseen based on their cluster embedding inside semantic space and WordNet hierarchy [64]. They have provided a list of 73,774 images containing only seen objects for training and 6,608 images containing both seen and unseen objects for testing. We adapt their exact setting to perform ZST and GZST. We compare our method with ConSE [46], Fast0Tag [14] and our baseline. Our approach constantly beats other state-of-the-art tagging methods in both the tasks. Note that, only 2,729 images inside the test set contains at least one unseen objects. As only seen objects dominate the test set, the GZST performance is higher than ZST for all the evaluated methods.

Top1 Accuracy	w2v	glo
Akata'16 [11]	33.90	-
DMaP-I'17 [9]	26.38	30.34
SCoRe'17 [10]	31.51	-
Akata'15 [65]	28.40	24.20
LATEM'16 [66]	31.80	32.50
Ours	**36.55**	**33.00**

TABLE VII: Zero-shot recognition on CUB using mean-pooling based MIL. For fairness, we only compared with the inductive setting of other methods without per image part annotation and description.

D. Zero Shot Recognition (ZSR)

Our proposed framework is designed to handle the zero-shot multi-label problem. Therefore, it can also be used for single label ZSR problem. To evaluate the performance on ZSR setting, we experiment with the Caltech-UCSD Birds-200-2011 (CUB) dataset [67]. Although the size of this dataset is relatively small containing 11,788 images belonging to 200 classes, it is popular for fine-grained recognition tasks. In ZSR literature [5], [66], the standard train/test split uses a fixed set of 150 seen and 50 unseen classes for experiments. We follow this traditional setting without using bounding boxes annotation, per image part annotation (like [11]) and descriptions (like [7]). To be consistent with the rest of the paper, we consider 400-d unsupervised GloVe (glo) and word2vec (w2v) vectors used in [66].

For a test image, our network predicts unseen class scores and an image is classified to the unseen class that gets the maximum score. As per standard practice, we report the mean Top1 accuracy of unseen classes in Table VII. Our method achieves superior results in comparison to state-of-the-art methods using the same settings. Note that all other methods are deep feature based approaches but do not train a joint framework in an end-to-end manner. In contrast, our method is end-to-end learnable based on ResNet-50 and additionally generates bounding boxes without using any box annotations.

E. Discussion

1) How does MIL help in multi-label zero-shot learning?: We explain this aspect using the illustration in Figure 5. One can observe that several tags pertain to localized information in a scene that is represented by only a small subset of the whole image, e.g., fish, coral, bike and bird. This demonstrates that a multi-label tagging method should consider localized regions in conjunction with the whole image. Our proposed method incorporates such a consideration using the MIL formulation. Therefore, it can annotate those localized tags where previous methods, e.g., Fast0tag, [14] usually fail (see rows 1-2 in Fig. 5). However, tags like beach, sunset, landscape in the third row of the figure are related to the global information in an image which does not depend on localized features. Therefore, in this respect, our method sometimes fails in comparison to Fast0tag [14] (see row 3 in Fig. 5). However, as illustrated in Fig. 5 (the non-bold tags in blue and black colors), the predicted tags of our method in those failure cases are still meaningful and relevant compared to the failure cases of Fast0tag [14].

2) Impact of pooling in MIL: The choice of pooling strategy as well as the domain that is used to perform feature aggregation, i.e., semantic domain (case 1) or visual domain (case 2) has a profound impact on zero-shot tagging. In our study, we find that pooling on visual features works better than pooling on prediction scores in the semantic domain. It tells us that pooling helps to reduce noise inside features more than noise in word vectors. Also, we observe than LSE outperforms mean, max and attention based pooling. It shows that LSE can achieve a good balance between the contributions of global and localized features.

3) Image location and tag correspondence: As a byproduct, our approach can generate a bounding box for each assigned tag. In Fig. 6, we illustrate some boxes (for top 2 tags) to indicate the correspondence between image locations and associated tags. Notably, our method often selects the whole image as one bounding box because we consider the entire

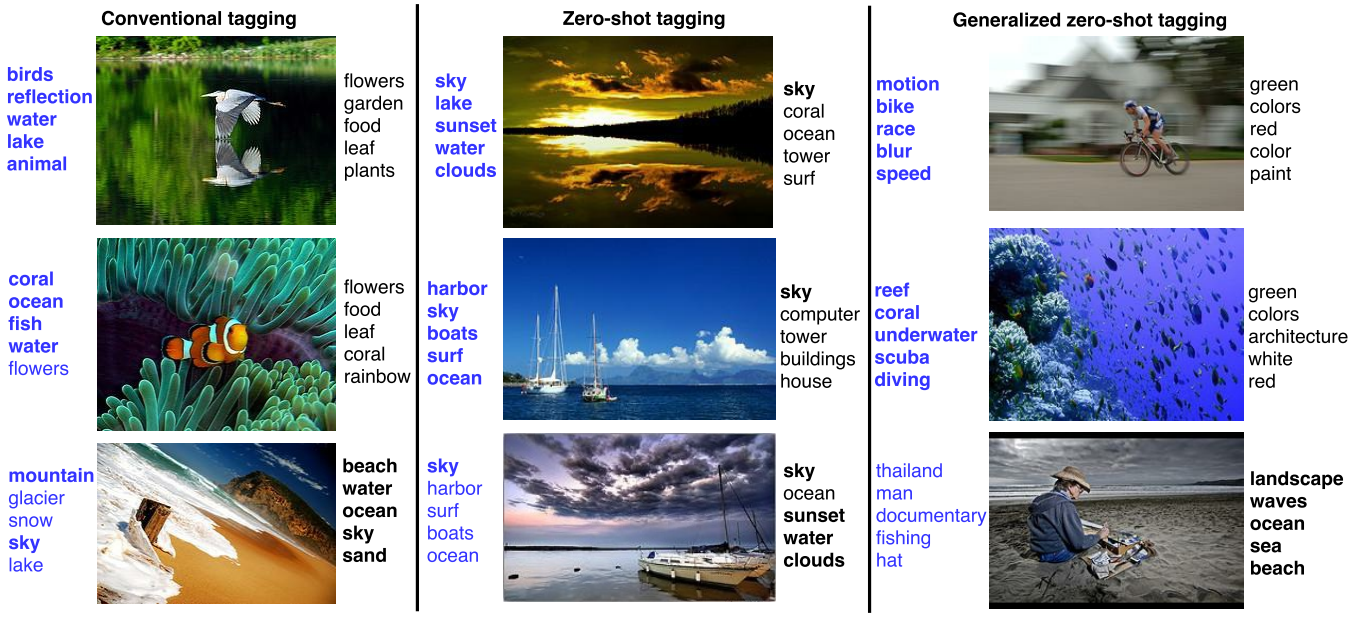

Fig. 5: Examples of top 5 predicted tags across different tasks by our method (left/blue) and Fast0tag [14] (right/black). **Bold** text represents the correct tags according to ground-truth. First two rows of images illustrate successful examples of our method and the third row is for negative cases.

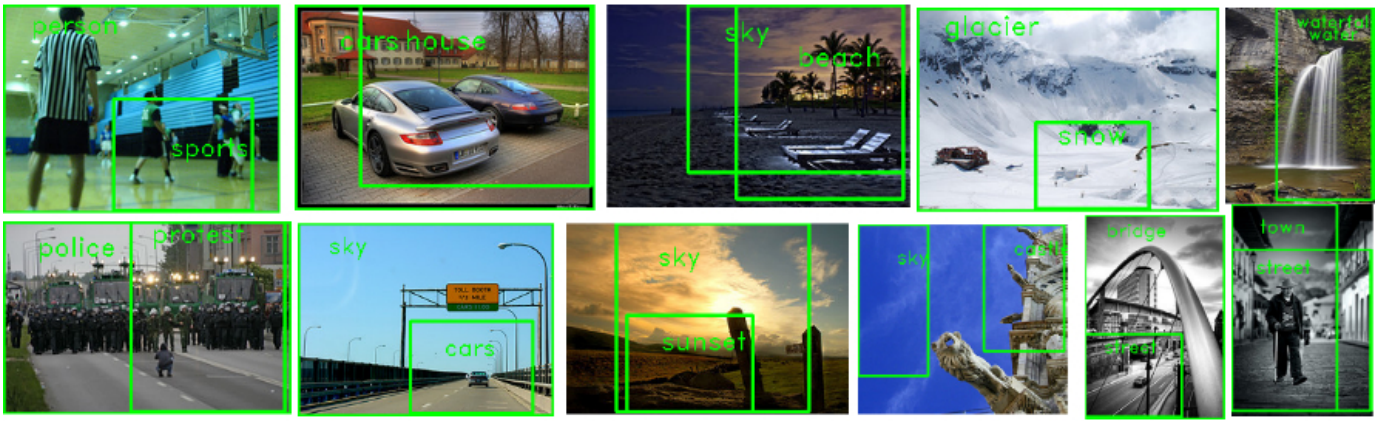

Fig. 6: Zero-shot tag discovery in natural images. Bounding boxes are shown for Top 2 tags in each image. Our approach not only assigns multiple tags to each image but also generates a bounding box for each tag.

image as an instance inside the bag. This consideration is particularly helpful for NUS-WIDE dataset because it contains many tags which are not only related to objects but are relevant to the overall scene, e.g., natural concepts (sky, water, sunset), aesthetic style (reflection, tattoo) or action (protest, earthquake, sports). Any quantitative analysis for this weakly supervised box detection task was not possible because the NUS-WIDE dataset does not provide any localization ground-truth for tags in an image.

V. Conclusion

While traditional zero-shot learning methods only handle a single unseen label per image, this paper attempts to assign multiple unseen tags. For the first time, we propose an end-to-end, deep MIL framework to tackle the multi-label zero-shot tagging problem. We integrate automatic patch discovery, feature aggregation and semantic domain projection within a single unified framework. Unlike previous models for traditional image tagging, our MIL framework does not depend on an off-line feature extraction and bag generation mechanisms. The proposed approach can inherently combine local as well as global scene details and can assign seen and/or unseen tags at test time. Moreover, any number of unseen tags from an open vocabulary could be used for prediction during test time. Our method can be viewed as a weakly supervised learning approach because of its ability to find a bounding box for each tag without requiring any box annotation during training. We validate our framework by achieving state-of-the-art performance on a large-scale tagging benchmark, outperforming established methods in the literature. As future work, the semantic relationship between word vectors can be explored to incorporate dependency among tags.

REFERENCES

[1] Li, Z., Tang, J.: Weakly supervised deep metric learning for community-contributed image retrieval. IEEE Transactions on Multimedia **17**(11) (2015) 1989–1999

[2] Wang, Z., Hu, R., Liang, C., Yu, Y., Jiang, J., Ye, M., Chen, J., Leng, Q.: Zero-shot person re-identification via cross-view consistency. IEEE Transactions on Multimedia **18**(2) (Feb 2016) 260–272

[3] Han, X., Singh, B., Morariu, V.I., Davis, L.S.: Vrfp: On-the-fly video retrieval using web images and fast fisher vector products. IEEE Transactions on Multimedia **19**(7) (July 2017) 1583–1595

[4] Cui, P., Liu, S., Zhu, W.: General knowledge embedded image representation learning. IEEE Transactions on Multimedia **20**(1) (Jan 2018) 198–207

[5] Xian, Y., Schiele, B., Akata, Z.: Zero-shot learning - the good, the bad and the ugly. In: CVPR. (2017)

[6] Kodirov, E., Xiang, T., Gong, S.: Semantic autoencoder for zero-shot learning. In: CVPR. (July 2017)

[7] Zhang, L., Xiang, T., Gong, S.: Learning a deep embedding model for zero-shot learning. In: CVPR. (July 2017)

[8] Deutsch, S., Kolouri, S., Kim, K., Owechko, Y., Soatto, S.: Zero shot learning via multi-scale manifold regularization. In: CVPR. (July 2017)

[9] Li, Y., Wang, D., Hu, H., Lin, Y., Zhuang, Y.: Zero-shot recognition using dual visual-semantic mapping paths. In: CVPR. (July 2017)

[10] Morgado, P., Vasconcelos, N.: Semantically consistent regularization for zero-shot recognition. In: CVPR. (July 2017)

[11] Akata, Z., Malinowski, M., Fritz, M., Schiele, B.: Multi-cue zero-shot learning with strong supervision. In: CVPR. (June 2016)

[12] Mensink, T., Gavves, E., Snoek, C.G.: Costa: Co-occurrence statistics for zero-shot classification. In: CVPR. (2014) 2441–2448

[13] Fu, Y., Yang, Y., Hospedales, T., Xiang, T., Gong, S.: Transductive multi-label zero-shot learning. (2014)

[14] Zhang, Y., Gong, B., Shah, M.: Fast zero-shot image tagging. In: CVPR. (June 2016)

[15] Li, X., Liao, S., Lan, W., Du, X., Yang, G.: Zero-shot image tagging by hierarchical semantic embedding. In: RDIR, ACM (2015) 879–882

[16] Wang, X., Zhu, Z., Yao, C., Bai, X.: Relaxed multiple-instance svm with application to object discovery. (2015) 1224–1232 cited By 10.

[17] Tang, P., Wang, X., Feng, B., Liu, W.: Learning multi-instance deep discriminative patterns for image classification. IEEE TIP **26**(7) (2017) 3385–3396

[18] Wu, J., Yu, Y., Huang, C., Yu, K.: Deep multiple instance learning for image classification and auto-annotation. In: CVPR. (June 2015) 3460–3469

[19] Wei, Y., Xia, W., Lin, M., Huang, J., Ni, B., Dong, J., Zhao, Y., Yan, S.: Hcp: A flexible cnn framework for multi-label image classification. IEEE TPAMI **38**(9) (2016) 1901–1907

[20] Tang, P., Wang, X., Huang, Z., Bai, X., Liu, W.: Deep patch learning for weakly supervised object classification and discovery. Pattern Recognition **71** (2017) 446 – 459

[21] Ren, Z., Jin, H., Lin, Z., Fang, C., Yuille, A.: Multiple instance visual-semantic embedding. In: BMVC. (2017)

[22] Girshick, R.: Fast r-cnn. In: ICCV. (December 2015)

[23] Uijlings, J.R.R., van de Sande, K.E.A., Gevers, T., Smeulders, A.W.M.: Selective search for object recognition. IJCV **104**(2) (Sep 2013) 154–171

[24] Zitnick, C.L., Dollár, P.: Edge boxes: Locating object proposals from edges. In Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T., eds.: ECCV, Cham, Springer International Publishing (2014) 391–405

[25] Cheng, M.M., Zhang, Z., Lin, W.Y., Torr, P.: Bing: Binarized normed gradients for objectness estimation at 300fps. In: CVPR. (2014) 3286–3293

[26] Rahman, S., Khan, S.: Deep multiple instance learning for zero-shot image tagging. In: Asian Conference on Computer Vision (ACCV). (December 2018)

[27] Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., Bengio, Y.: Show, attend and tell: Neural image caption generation with visual attention. In: International conference on machine learning. (2015) 2048–2057

[28] Pappas, N., Popescu-Belis, A.: Explicit document modeling through weighted multiple-instance learning. Journal of Artificial Intelligence Research **58** (2017) 591–626

[29] Quellec, G., Cazuguel, G., Cochener, B., Lamard, M.: Multiple-instance learning for medical image and video analysis. IEEE Reviews in Biomedical Engineering **10** (2017) 213–234

[30] Ramon, J., De Raedt, L.: Multi instance neural networks. In: Proceedings of the ICML-2000 workshop on attribute-value and relational learning. (2000) 53–60

[31] Pinheiro, P.O., Collobert, R.: From image-level to pixel-level labeling with convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 1713–1721

[32] Ilse, M., Tomczak, J., Welling, M.: Attention-based deep multiple instance learning. In Dy, J., Krause, A., eds.: Proceedings of the 35th International Conference on Machine Learning. Volume 80 of Proceedings of Machine Learning Research., StockholmsmÃd'ssan, Stockholm Sweden, PMLR (10–15 Jul 2018) 2127–2136

[33] Durand, T., Mordan, T., Thome, N., Cord, M.: Wildcat: Weakly supervised learning of deep convnets for image classification, pointwise localization and segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2017) 642–651

[34] Demirel, B., Gokberk Cinbis, R., Ikizler-Cinbis, N.: Attributes2classname: A discriminative model for attribute-based unsupervised zero-shot learning. In: ICCV. (Oct 2017)

[35] Rahman, S., Khan, S., Porikli, F.: A unified approach for conventional zero-shot, generalized zero-shot, and few-shot learning. IEEE Transactions on Image Processing **27**(11) (Nov 2018) 5652–5667

[36] Rahman, S., Khan, S., Porikli, F.: Zero-shot object detection: Learning to simultaneously recognize and localize novel concepts. In: Asian Conference on Computer Vision (ACCV). (December 2018)

[37] Bansal, A., Sikka, K., Sharma, G., Chellappa, R., Divakaran, A.: Zero-shot object detection. In: The European Conference on Computer Vision (ECCV). (September 2018)

[38] Rahman, S., Khan, S., Barnes, N.: Polarity loss for zero-shot object detection. arXiv preprint arXiv:1811.08982 (2018)

[39] Li, J., Wei, Y., Liang, X., Dong, J., Xu, T., Feng, J., Yan, S.: Attentive contexts for object detection. IEEE Transactions on Multimedia **19**(5) (2017) 944–954

[40] Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. IEEE TPAMI **39**(6) (June 2017) 1137–1149

[41] Jifeng Dai, Yi Li, K.H.J.S.: R-FCN: Object detection via region-based fully convolutional networks. arXiv preprint arXiv:1605.06409 (2016)

[42] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C. In: SSD: Single Shot MultiBox Detector. Springer International Publishing (2016) 21–37

[43] Redmon, J., Farhadi, A.: Yolo9000: Better, faster, stronger. arXiv preprint arXiv:1612.08242 (2016)

[44] Lin, T.Y., Goyal, P., Girshick, R., He, K., Dollar, P.: Focal loss for dense object detection. In: The IEEE International Conference on Computer Vision (ICCV). (Oct 2017)

[45] Lin, T.Y., Dollar, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (July 2017)

[46] Norouzi, M., Mikolov, T., Bengio, S., Singer, Y., Shlens, J., Frome, A., Corrado, G.S., Dean, J.: Zero-shot learning by convex combination of semantic embeddings. In: ICLR. (2014)

[47] Lee, C.W., Fang, W., Yeh, C.K., Frank Wang, Y.C.: Multi-label zero-shot learning with structured knowledge graphs. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (June 2018)

[48] He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. Volume 2016-January. (2016) 770–778 cited By 107.

[49] Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)

[50] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. CVPR **07-12-June-2015** (2015) 1–9

[51] Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. CVPR **1**(2) (2017) 4

[52] Bourbaki, N.: Eléments de mathématiques: théorie des ensembles, chapitres 1 à 4. Volume 1. Masson (1990)

[53] Zaheer, M., Kottur, S., Ravanbakhsh, S., Poczos, B., Salakhutdinov, R.R., Smola, A.J.: Deep sets. In: NIPS. (2017) 3391–3401

[54] Hassoun, M.H.: Fundamentals of artificial neural networks. MIT press (1995)

[55] Li, Y., Song, Y., Luo, J.: Improving pairwise ranking for multi-label image classification. In: Proceedings of the IEEE conference on computer vision and pattern recognition. (2017) 3617–3625

[56] Chua, T.S., Tang, J., Hong, R., Li, H., Luo, Z., Zheng, Y.T.: Nus-wide: A real-world web image database from national university of singapore. In: CIVR, Santorini, Greece. (July 8-10, 2009)

[57] Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: EMNLP. (2014) 1532–1543

[58] Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: NIPS. (2013) 3111–3119
[59] Gong, Y., Jia, Y., Leung, T., Toshev, A., Ioffe, S.: Deep convolutional ranking for multilabel image annotation. arXiv preprint arXiv:1312.4894 (2013)
[60] Weston, J., Bengio, S., Usunier, N.: Wsabie: Scaling up to large vocabulary image annotation. (2011) 2764–2770
[61] Guillaumin, M., Mensink, T., Verbeek, J., Schmid, C.: Tagprop: Discriminative metric learning in nearest neighbor models for image auto-annotation. In: ICCV. (2009) 309–316
[62] Chen, M., Zheng, A., Weinberger, K.Q.: Fast image tagging. In: ICML, ICML (January 2013)
[63] Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: ECCV, Springer (2014) 740–755
[64] Miller, G.A.: Wordnet: a lexical database for english. Communications of the ACM **38**(11) (1995) 39–41
[65] Akata, Z., Reed, S., Walter, D., Lee, H., Schiele, B.: Evaluation of output embeddings for fine-grained image classification. In: CVPR. Volume 07-12-June-2015. (2015) 2927–2936
[66] Xian, Y., Akata, Z., Sharma, G., Nguyen, Q., Hein, M., Schiele, B.: Latent embeddings for zero-shot classification. In: CVPR. (June 2016)
[67] Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology (2011)

Shafin Rahman received a Bachelor degree in Computer Science and Engineering from Bangladesh University of Engineering and Technology (BUET), in March 20011, and a Master degree in Computer Science from University of Manitoba, Winnipeg, Canada, in October 2015. Since 2016, he is a Ph.D. student at the Australian National University (ANU) and Data61, Commonwealth Scientific and Industrial Research Organization. Previously, he served North South University, Bangladesh as a full-time lecturer. He also worked as a senior software engineer at Samsung R&D Institute, Bangladesh. His primary research interests are visual saliency, scene understanding by connecting vision and language with a specific focus on zero-shot learning.

Salman Khan received the B.E. degree in electrical engineering from the National University of Sciences and Technology, Pakistan, in 2012, and the Ph.D. degree from The University of Western Australia, in 2016. His Ph.D. thesis received an honorable mention on the Dean's List Award. He was a Visiting Researcher with National ICT Australia, CRL, in 2015. From 2016 to 2018, he was a Research Scientist with Data61, Commonwealth Scientific and Industrial Research Organization. He has been a Senior Scientist with Inception Institute of Artificial Intelligence, since 2018, and an Adjunct Lecturer with Australian National University, since 2016. He was a recipient of several prestigious scholarships, including Fulbright and IPRS. He has served as a program committee member for several premier conferences, including CVPR, ICCV, and ECCV. In 2019, he was awarded the outstanding reviewer award at CVPR. His research interests include computer vision, pattern recognition, and machine learning.

Nick Barnes received the B.Sc. (Hons.) and Ph.D. degrees in computer vision for robot guidance from The University of Melbourne, Australia, in 1992 and 1999, respectively. In 1999, he was a Visiting Research Fellow with the LIRA Laboratory, University of Genova, Italy. From 2000 to 2003, he was a Lecturer with the Department of Computer Science and Software Engineering, The University of Melbourne. Since 2003, he has been with the NICTA's Canberra Research Laboratory, which as become Data61, Commonwealth Scientific and Industrial Research Organization, Australia, where he is currently a Senior Principal Researcher and leads Computer Vision. He has also been an Associate Professor at the Australian National University. His current research interests include dense estimation tasks in computer vision, as well as prosthetic vision, biologically inspired vision, and vision for vehicle guidance. He has published more than 140 research papers on these topics.