Zero-shot Learning Using Multimodal Descriptions

Utkarsh Mall, Bharath Hariharan, Kavita Bala
Cornell University
{utkarshm, bharathh, kb}@cs.cornell.edu

Abstract

Zero-shot learning (ZSL) tackles the problem of recognizing unseen classes using only semantic descriptions, e.g., attributes. Current zero-shot learning techniques all assume that a single vector of attributes suffices to describe each category. We show that this assumption is incorrect. Many classes in real world problems have multiple modes of appearance: male and female birds vary in appearance, for instance. Domain experts know this and can provide attribute descriptions of the chief modes of appearance for each class. Motivated by this, we propose the task of multimodal zero-shot learning, where the learner must learn from these multimodal attribute descriptions. We present a technique for addressing this problem of multimodal ZSL that outperforms the unimodal counterpart significantly. We posit that multimodal ZSL is more practical for real-world problems where complex intra-class variation is common.

1. Introduction

Recognition systems today are accurate but rely on large training sets. In many domains, for example, in medical imagery or scientific domains, acquiring such large scale training sets is difficult. Images must be labeled by an expert, whose time is valuable. Further, acquiring the images themselves can sometimes be difficult, as for X-ray images in medicine. This has motivated research into alternatives that use limited supervision. One alternative that is reminiscent of how humans learn from experts is offered by *zero-shot learning*. Here, an expert specifies a new class simply by providing a semantic description of the class, for instance in terms of its *attributes*: a grey heron has a "long beak", "black crown" and "grey body". The system must then learn to recognize this new class using these descriptions alone.

While many forms of semantic descriptions can be used, the most accurate zero-shot learning descriptions use vectors of attributes as semantic class descriptions [1, 21, 7, 23, 6, 31, 2, 26, 22, 16]. However, all attribute-based ZSL work makes one key assumption: that *every class can be described with a single vector of attributes*. Unfortunately, this assumption is not true in practice, because objects in a class can vary dramatically in appearance, enough to have very different attributes. For example, Figure 1 shows all the many variations of an American Goldfinch. The body can be a bright yellow in breeding males, a dull yellow in breeding females, brown in non-breeding males or pale yellow in immature birds. Even among breeding males, some males may have large streaks of white in their wings while others don't. Breeding males have a black cap, while the cap is olive in breeding females. Similar variations arise in other domains as well. For example, in scene classification, a badminton court can have different appearance based on the playing surface (see Figure 3) (bottom right). Describing all these variations with a single attribute vector as is done in past work would either miss most of these modes of appearance entirely, or result in an "average" attribute vector that is too diffuse to be discriminative. The resulting classifiers are thus doomed to be inaccurate.

Figure 1. The many faces of an American Goldfinch. From left to right: Breeding male, breeding male, breeding female, immature, immature/female and nonbreeding-male.

While the entire distribution of appearance may be hard to capture using a few attribute specifications, we posit that an annotator with domain knowledge would know the chief *modes* of appearance and can specify them. For example an ornithologist can tell us what a breeding male, breeding female or a juvenile American Goldfinch would *typically* look like. Indeed, the many modes of appearance of the American Goldfinch shown in Figure 1 were derived from an online field guide about birds. However, current zero-shot

learning techniques offer no way of learning from such a specification. What is needed is a new family of zero-shot learning techniques that can handle multimodal attribute descriptions.

In this paper, we formalize this more practical zero-shot learning problem, which we call *multimodal zero-shot learning*. We then propose a new zero-shot learning technique that can leverage these multimodal attribute annotations. Our technique generalizes prior work and can be applied to multiple prior unimodal zero-shot learning methods. We evaluate our approach on multimodal ZSL benchmarks that we create using 3 existing attribute-based datasets commonly used for ZSL: CUB [27], SUN [18] and DeepFashion [13]. We show that reasoning about multiple modes can provide a significant improvement in accuracy ($\sim 5\%$ improvement on CUB). Our contributions are:

1. We show that existing zero-shot learning techniques suffer by assuming a single attribute description for each class, and we identify a need for descriptions that specify multiple modes.
2. We define a new *multimodal zero-shot learning* task.
3. We present a new multimodal zero-shot learning technique that can dramatically improve accuracy by reasoning about multiple modes.

2. Related work

In zero-shot learning [10, 11], the model is given a set of training classes and test classes. The model has access to images in the training set and is also given side information, like attribute descriptions for both the training and test classes. The performance of models is judged by how well they classify images from the test classes just using the information about the new classes. Initial work by Lampert *et al*. [10] proposed first predicting attributes from images and then classifying images based on predicted attributes. With the discovery that convolutional networks tend to learn generalizable feature representations, a series of papers have looked at projecting image features into attribute space, and measuring similarity with class descriptions [5, 1, 21, 7, 24].

However, using attribute descriptions directly as a featurization of a class might be a naive approach as the discriminability of different attributes is unclear. Some work proposes to learn in the reverse direction, by featurizing classes in the visual feature space and learn a mapping from attributes to images [23, 6].

As such, more recent work has tended to use the attribute description to produce a more discriminative class embedding. The general idea is to embed these descriptions as well as images into a shared feature space [31, 2, 26, 17] where classification is as easy as computing the inner product similarity or euclidean distance. This shared space can be trained to optimize a classification loss on some "base classes" with labeled examples, although auxilliary losses such as a reconstruction loss can be used to regularize the problem [24]. Xian et al. provide a comprehensive survey of these and other techniques [29]. Using generative models for representation learning has lead to many new works outperforming the discriminative learning models. The models have successfully used adversarial learning [30], variational auto-encoders[15, 8, 22], or VAEGANs[31, 16] to learn a generative representation and then use it for zero-shot learning.

In this work we propose a new problem of learning from multimodal attributes. We also show how our method can use any zero-shot learning models where the attributes and images are encoded into a common latent space. With multimodal attributes and our method, we surpass the corresponding models that use unimodal attributes.

Several popular benchmarks have been proposed for zero-shot learning in the past. These datasets have classes with images and an associated attribute description. Lampert *et al*. proposed Animals with Attributes dataset (AwA)[9] consisting of 50 animal categories. Zero-shot benchmarks for other domains are also proposed, including scenes (SUN) [18], birds (CUB)[27] and common objects (aPascal/aYahoo) [4]. While these datasets contain rich attribute information for classes, all of described classes use a *single* fractional attribute descriptor. We instead create new benchmarks with multiple attributes descriptors capturing different modes of variation as described in Sec. 4.2 and supplementary.

Several methods have been proposed for zero-shot learning from text descriptions such as Wikipedia articles [19, 32, 3, 2, 20] or learning from partial attributes [14]. While such descriptions can potentially describe multiple modes of a class efficiently, existing methods model them as a single mode. This is partly due to the fact that text is hard to interpret for the learner. As a result these methods have significantly lower performance than the attribute based counterparts. Some zero-shot learning methods also try to implicitly model the intra-class variance in image space[28]. But unlike our work they assume zero-shot description does not describe this variance. As such, the unimodal zero-shot descriptions used during inference could mislead a classifier. Our work is the first to note that zero-shot description itself can include modes of variations.

3. Method

3.1. Problem Setup

We first describe the traditional (unimodal) zero-shot learning setup, and then show how we generalize it to the multimodal case.

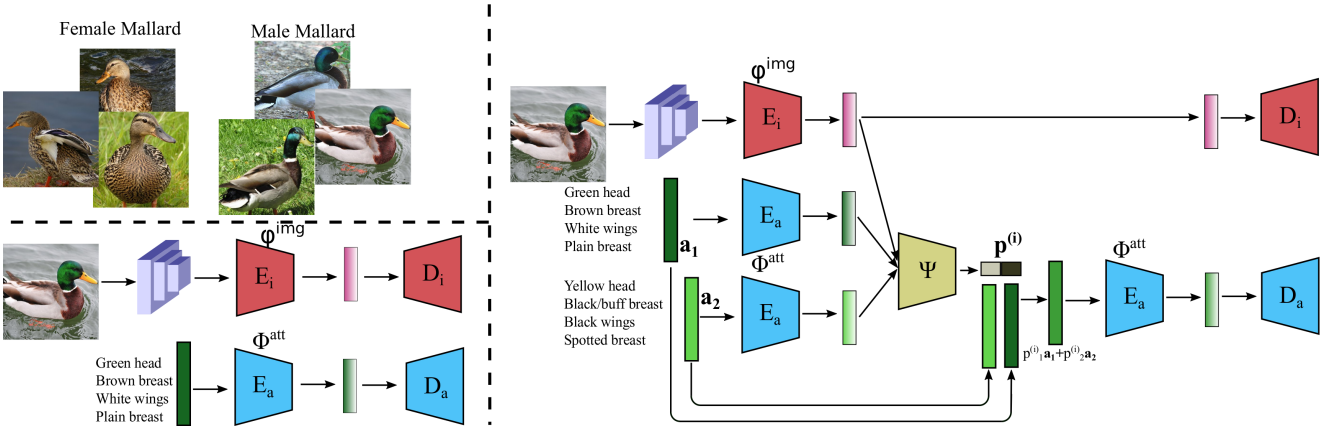

Figure 2. Overview of our method and new assignment module Ψ. **Top left**: Mallard with 2 primary modes of appearance (male and female) that look very different. **Bottom left**: CADA-VAE model with 2 VAEs aligning the attribute embedding and image embedding with the loss function L. Without our method, CADA-VAE and other similar models can only use a single attribute descriptor for the Mallard, irrespective of whether the bird in the image is male of female. **Right**: Modified CADA-VAE model to handle multimodal attributes. The assignment module Ψ uses the image embedding and attribute embedding to find the probability distribution $\mathbf{p}^{(i)}$, and then interpolates the attributes. As training progresses, the assignment module improves in finding the correct mode assignment.

Unimodal Zero-shot Learning. Zero-shot learning methods, typically run in two phases: a *representation learning* phase and a *deployment* phase. During the representation learning phase, the system (or learner) must learn how to map attribute vectors to class distinctions. It does so on a set of *base (seen) classes* $\mathcal{B}$ for which both images and attribute descriptions are available. Thus, the learner has a large labeled *training set* D consisting of images $\mathbf{x}_i, i = 1, \ldots, n$ and corresponding labels y_i. In addition, for each base class $y \in \mathcal{B}$, the learner is provided with a vector of attributes $\mathbf{a}(y)$.

Once trained, the learner is *deployed*. It now gets a set of hitherto novel (unseen) classes $\mathcal{N}$ for which the only available information is the corresponding vector of attributes. The learner must then learn to recognize these unseen classes.

As discussed above, this problem setup assumes that a single attribute vector suffices to describe each class. We now extend this setup to handle multimodal attribute descriptions.

Multimodal Zero-shot Learning. In multimodal zero-shot learning (MZSL), for each class y, be they base or novel, the learner gets a *set* of attribute vectors $A(y) = \{\mathbf{a}_1, \ldots, \mathbf{a}_m\}$, each of which corresponds to a particular mode of appearance.

Having such multimodal attributes for each class provides additional information for learning. However, properly leveraging this information requires solving additional challenges. Because each class can have multiple modes of appearance, an image of the class will only conform to *one* of the attribute vectors. However, during representation learning, the learner does not know which image conforms to which mode. This poses a challenge in representation learning, making it difficult for the learner to learn the visual signature of each attribute.

Below we describe our proposed approach which solves this challenge and learns effectively from multimodal descriptions.

3.2. Learning from Multimodal Attributes

Background: We build upon zero-shot learning methods that learn a common embedding space for attribute descriptions and images. This is done in a variety of different ways, ranging from linear mappings of image features to attributes [21, 7], to learning the embedding using adversarial learning [30, 16] and variational autoencoders [22]. In each case, there is an image encoder ϕ^{img} and an attribute encoder ϕ^{att}. In the *representation learning phase*, these encoders are obtained by optimizing some representation learning loss L:

$$\min L(\phi^{img}, \phi^{att}, \{(\mathbf{x}_i, y_i, \mathbf{a}(y_i)\}_{i=1}^{n}) \quad (1)$$

In general, L pushes the image embedding $\phi^{img}(\mathbf{x}_i)$ to align with the embedding of the corresponding attribute vector, $\phi^{att}(\mathbf{a}(y_i))$.

After the model learns a representation, it is *deployed*, and must now train classifiers for novel classes for which only attribute descriptions are available. One typically uses the attribute encoder ϕ^{att} to embed the description into latent space, and then uses the embedded vectors to train a classifier. Zero-shot methods can employ a variety of techniques to build these classifiers, such as nearest neighbor classifiers[6], SVMs [1] or linear classifiers [22].

We now describe how we adapt these techniques to the multimodal benchmark. We describe in turn the representation learning phase and the deployment phase.

3.2.1 Representation learning

In the multimodal setting, now for each class y we have a *set* of attribute vectors $A(y)$. Unfortunately, we do not know which image belongs to which mode. To remedy this, we introduce an assignment module, Ψ, that learns the assignment of an image to its corresponding mode during representation learning.

Concretely, consider an image $\mathbf{x}_i$ with its corresponding label y_i. Suppose the set of attribute vectors for this class is $A(y_i) = \{\mathbf{a}_1, \ldots, \mathbf{a}_m\}$. Denote by $\Phi^{att}(y_i)$ the corresponding attribute embeddings: $\Phi^{att}(y_i) = \{\phi^{att}(\mathbf{a}_i), \ldots, \phi^{att}(\mathbf{a}_m)\}$. The assignment module takes these attribute embeddings as input along with the image embedding, and produces a probability distribution $\mathbf{p}^{(i)}$ of the image belonging to each of the different modes:

$$\mathbf{p}^{(i)} = \Psi(\phi^{img}(x_i), \Phi^{att}(y_i)) \tag{2}$$

Using these predicted probabilities as weights, we construct an attribute vector $\mathbf{a}^{(i)}$ for this image:

$$\mathbf{a}^{(i)} = \sum_{j=1}^{m} p_j^{(i)} \mathbf{a}_j \tag{3}$$

Our final training objective then simply slots in these estimated attribute values into Equation (1):

$$L(\phi^{img}, \phi^{att}, \{(\mathbf{x}_i, y_i, \mathbf{a}^{(i)}\}_{i=1}^{n}) \tag{4}$$

The assignment module is trained along with the zero-shot model on the latent space of the model. This loss function optimizes both, the assignment module to produce better assignments and the zero-shot latent space to align correct modes to the image. We next look at the architectural design and the training procedure for the assignment module.

Assignment Module. The assignment module takes the latent embedding of multimodal attribute descriptions and images and uses it to produce an assignment probability. We take advantage of the fact that these embeddings are being trained to align the image and attribute spaces, and propose to simply use the cosine similarity between attribute and image embeddings to make the assignment. Denoting the cosine similarity between vectors $\mathbf{a}$ and $\mathbf{b}$ as $\langle \mathbf{a}, \mathbf{b} \rangle$, our assignment module takes the form:

$$\Psi(\phi^{img}(\mathbf{x}_i), \Phi^{att}(y_i))[j] = \frac{e^{\langle \phi^{img}(\mathbf{x}_i), \phi^{att}(\mathbf{a}_j) \rangle / T}}{\sum_{\mathbf{a}_k \in A(y_i)} e^{\langle \phi^{img}(\mathbf{x}_i), \phi^{att}(\mathbf{a}_k) \rangle / T}} \tag{5}$$

Here T is a temperature: higher values lead to softer probability distributions. A challenge here is that this formulation relies on good embeddings to match images to attribute descriptions. But at the start of training, such good embeddings do not exist. As such, the predicted probabilities might be extremely noisy. To address this challenge, we divide the learning into 3 stages:

1. For the first third of the training epochs, we set the temperature to ∞ so that predicted probabilities are uniform. This leads to every image being assigned an attribute vector that is the *mean* of the provided modes. While sub-optimal, this allows us to bootstrap our embeddings with what is essentially *unimodal* training. Enough classes have few enough modes that this still results in reasonable initial embeddings.

2. For the next third of the total training epochs, we decrease the temperature asymptotically (reciprocal of temperature is increased linearly). This slowly forces the model to commit to a single mode for each image.

3. Finally, in the last third of training we replace soft probabilities with one-hot vectors obtained through an argmax, so that at convergence, each image is assigned to one and only one mode.

Note that argmax is not differentiable. In fact, even in the earlier stages of training when the assignment module is differentiable, we do not allow gradients to flow through the assignment module. We found that this makes the training both more stable and faster.

Our framework (eq. 1-5) can be applied to any existing or future zero-shot learning model. For CADA-VAE[22] as the base learner, Figure 2 shows the side-by-side comparison of the original CADA-VAE model and our modified multimodal version with the assignment module.

Other potential formulations. Observe that the output of the assignment module is used to interpolate between attribute vectors. It is also possible to interpolate in the latent space instead and modify the loss accordingly. Interestingly, this choice has a dramatic negative impact on performance (4.5% drop on CUB), and it is crucial that the interpolation be done in attribute space. This hints towards significant differences in the semantics of the attribute space and the latent space.

The assignment module in itself is non-parametric, but as it works in the latent space of the zero-shot model it affects the parameters of zero-shot model when training. We also tried experiments with a parametric assignment module such as with weighted distances, but preliminary experiments suggested that the non-parametric architecture works best.

3.2.2 Deployment

Once trained, the learner will be deployed. Again, for each novel class y, it will get a set of attribute descriptions $A(y) = \{\mathbf{a}_1, \ldots, \mathbf{a}_m\}$. The learner will now use Φ^{att} to embed each

of these descriptions, yielding a *set* of embedded vectors for each class. These sets can then be used to train a classifier in the embedding space as before. As such the model is in principle similar to a unimodal zero-shot model during deployment.

However, in practice, traditional zero-shot models use linear classifiers in the latent space, assuming that classes are linearly separable in the learned representation. This works for the unimodal case because each class is represented by a single point in the embedding space. But with multimodal attributes, the different modes for a class may be very different and thus the resulting collection of embedded vectors may not be cleanly separable from other classes. Hence, we also consider non-linear classifiers . In experiments, we present the results with the best classifiers for each method below.

4. Experimental setup

4.1. Base learner

Our method can be used with any zero-shot model out-of the box. In our experiments, we explore two different base learners: CADA-VAE[22] and TF-VAEGAN[16]. CADA-VAE uses two variational auto-encoders to learn a common embedding space for attribute descriptions and images. It trains these autoencoders on base class examples during representation learning. In deployment it trains a classifier with the latent novel attribute encoding as inputs to learn about novel classes. TF-VAEGAN uses a VAE-GAN [12] to generate realistic features from attributes and uses these generated features from unseen classes to train a classifier. For fairness in comparisons we use the same model hyperparameters and training parameters for all the methods as in their original implementation.

4.2. Benchmarks for Multimodal ZSL

In traditional zero-shot learning benchmarks every class (base or novel) is associated with a single attribute description. Instead, our proposed method allows and makes use of multiple attribute descriptions per class. Evaluating the promise of this approach requires benchmarks with multiple attribute descriptions for each class. We created such multimodal ZSL benchmarks from 3 existing datasets: CUB-200-2011 [27] (CUB), SUN attributes [18] (SUN) and DeepFashion[13] (DF).

CUB is annotated with 312 part-based attributes, whereas SUN has 102 attributes. DeepFashion has 1000 attributes, but these are very noisy and unreliable (in fact very few attributes are consistent across a class). CUB and SUN are commonly used in ZSL research, and we use the splits proposed by Xian *et al*. [29]. Since DF is a new dataset for this task, we filter to only include the clothing categories with more than 300 instances. This results in 33 categories that we split into 25 base classes and 8 novel classes.

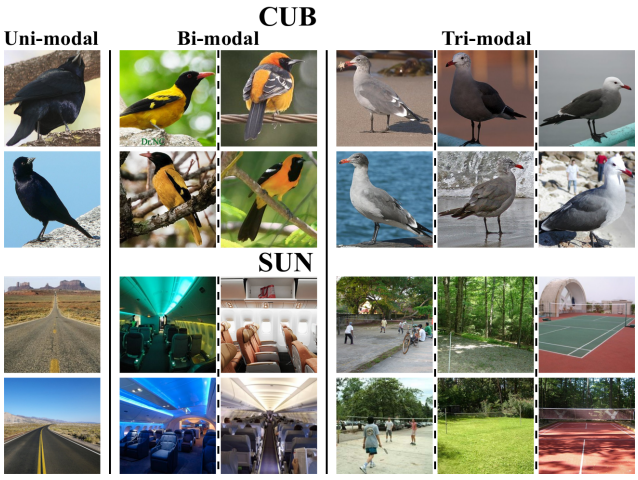

Figure 3. The rows show the modes of attributes for different classes of CUB (top 2 rows) found by manual grouping and SUN (bottom 2 rows) found using automatic grouping. The leftmost column shows unimodal classes "Shiny Cowbird" and "Desert Road". The center two columns show bimodal classes "Hooded Oriole" and "Airplane Interior"; and the rightmost 3 columns show trimodal classes "Heerman Gull" and "Badminton Court Outdoor".

In all 3 benchmarks, a class is annotated with multiple modes of attribute descriptions that are intended to capture the different modes of appearance. Different classes can have different number of modes based on the different variations it can have. Figure 3 shows the modes for several classes in the multimodal benchmark for CUB and SUN dataset. Some classes are unimodal for example "Shiny Cowbird" and "Desert Road" (left column). Others require multimodal attribute descriptions, 2 modes: "Hooded Oriole" and "Airplane Interior" (center 2 columns) and 3 modes "Heerman Gull" and "Badminton Court Outdoor" (center two columns). While the figure shows the images belonging to these modes, the attribute vector also has differences. For example, the modes of Hooded Oriole differ significantly in their attributes "color (yellow and orange)", "forehead color (black and orange)", and "bill color (orange and black)".

We explored two ways of creating these multimodal annotation. In the "manual" approach, we asked annotators for modes of variation, whereas in the "automatic" approach, the modes were discovered automatically, using the same raw data that is used in the traditional ZSL benchmarks. Both benchmarks will be released upon acceptance. More details on how we create these benchmarks are presented in the **supplementary**; we focus here on the approach itself.

Binary vs. Real-valued Attributes. The traditional ZSL benchmarks consider both binary and real-valued attribute descriptors. We follow past work and similarly consider both binary and real-valued attribute descriptors.

However, we note that obtaining real-valued attribute

Method	Real-valued	Binary
MZSL (automatic)	$\mathbf{60.4 \pm 0.3}$	42.4 ± 0.2
MZSL (manual)	$\mathbf{60.2 \pm 0.4}$	$\mathbf{44.0 \pm 0.3}$

Table 1. Performance of MZSL in top-1 unseen class accuracy when the multimodal data is collected automatically vs when it is collected manually for CUB with CADA-VAE. Manual collection leads to a 1.6% gain in accuracy in the binary setting.

descriptors for novel classes is practically unrealistic, since it is difficult for human experts to provide precise floating point values. In contrast a binary-valued attribute descriptor is easy to provide. In our experiments, we find that multimodal descriptions lead to greater improvement for binary attribute descriptors and are thus more instrumental in a practical setting. Nonetheless we evaluate our method on both settings. Details for obtaining binary and real-valued attributes are present in the supplementary.

Manual vs Automatic Benchmarks One might be concerned that the automatically created benchmark may not match expert judgments of the modes of appearance of a class, and as such may yield incorrect results. We therefore did a first experiment to ensure that the two benchmarks yielded consistent results. Table 1 shows the zero-shot classification performance of our approach on unseen classes on CUB, on the manual and automatic benchmarks. Manual correction of the modes has a fairly small impact: a 1.6% improvement in the binary setting, and no change in the real-valued setting. This suggests that the automatically produced multimodal benchmark works almost as well as manual curation. As such, for SUN and DF, we only use the automatically created benchmarks. For CUB, all experiments use the manually collected benchmarks.

5. Results

In this section we show that using multimodal attributes indeed helps in improving the performance of the method. We first present quantitative results demonstrating the superiority of multimodal ZSL (MZSL) over traditional unimodal ZSL (UZSL). We then compare our MZSL method to other possible baseline approaches for multimodal ZSL. Finally, we also present some qualitative visualizations of our learned embeddings.

All results are averaged over 6 different runs of the model.

5.1. Are multimodal descriptions more helpful than a unimodal description?

We first evaluate the performance of our MZSL against the UZSL. Table 2 shows the performance (top-1 per class unseen accuracy) of our method MZSL when compared against zero-shot learning with unimodal descriptions with CADA-VAE. Having multimodal attributes leads to improvements

Real-valued Attributes			
Method	**CUB**	**SUN**	**DF**
UZSL	58.8 ± 0.2	59.4 ± 0.4	58.0 ± 0.3
MZSL	$\mathbf{60.2 \pm 0.4}$	$\mathbf{62.0 \pm 0.3}$	$\mathbf{58.3 \pm 0.4}$
Binary Attributes			
UZSL	38.9 ± 0.3	40.5 ± 0.2	46.4 ± 0.8
MZSL	$\mathbf{44.0 \pm 0.3}$	$\mathbf{47.0 \pm 0.3}$	$\mathbf{47.4 \pm 0.7}$

Table 2. Comparison of UZSL with MZSL on all datasets with real-valued and binary setup and CADA-VAE as base model. Results show top-1 per class unseen accuracy averaged over 6 runs. Our method MZSL performs better than its unimodal counterpart.

	TF-VAEGAN	
	Real-valued	**Binary**
UZSL	64.7 ± 0.2	42.4 ± 0.1
MZSL	$\mathbf{65.6 \pm 0.1}$	$\mathbf{45.9 \pm 0.2}$

Table 3. Comparison of UZSL with MZSL on CUB dataset with **TF-VAEGAN** as the base learner, with real-valued and binary attributes.

in performance with both real-valued and binary attributes on all three datasets. The improvement is larger when using binary attributes (5.1% improvement for CUB and 6.5% for SUN). As discussed in Section 4.2, binary attributes are more akin to what the learner might get in practice, since the annotator will not have access to multiple images to average attribute values. Having multimodal attributes is especially beneficial in this practical and realistic case.

As stated before, our method could be applied to any model that has a common latent embedding space for images and attributes. So we also compare the performance of MZSL with TF-VAEGAN as the base model. Table 3 shows the performance of our method MZSL on CUB with TF-VAEGAN. In the binary setting our method again leads to an improvement of 3.5% over unimodal attributes. This shows that our method is generalizable to other backbone models as well. Performance of TF-VAEGAN on other datasets is presented in the supplementary.

We also evaluate the performance of our approach for Generalized ZSL (GZSL), where the learner is evaluated on both seen and unseen classes, and performance is measured using the harmonic mean of seen and unseen accuracy. Table 4 shows the GZSL performance of MZSL for both CADA-VAE and TF-VAEGAN. MZSL outperforms unimodal descriptions in the GZSL setting as well. TF-VAEGAN is the state-of-the-art model for GZSL with real-valued attributes (on the proposed splits and features used by [29]). Our multimodal approach achieves a higher accuracy ($\mathbf{58.9}$%) thus yielding a new state-of-the-art.

	CADA-VAE		TF-VAEGAN	
	Real valued	Binary	Real valued	Binary
UZSL	52.1 ± 0.3	33.8 ± 0.4	57.8 ± 0.1	38.4 ± 0.2
MZSL	$\mathbf{53.2 \pm 0.4}$	$\mathbf{37.8 \pm 0.4}$	$\mathbf{58.9 \pm 0.1}$	$\mathbf{41.9 \pm 0.1}$

Table 4. Generalized ZSL metrics on the CUB dataset, for both the CADA-VAE and TF-VAEGAN-based models.

5.2. Can unimodal methods leverage multimodal descriptions?

Since there is no prior work that uses multimodal descriptions, we construct baselines that reduce multimodal ZSL into a unimodal ZSL problem. More specifically we look at the following four baselines:

- **Mean of Modes.** This baseline aggregates the modes by averaging together the attribute vectors of all the modes for each class. This single attribute vector is used during both training and deployment as in UZSL.
- **Weighted Mean of Modes.** Instead of a simple average, this baseline uses a weighted average of the attribute vectors, with the weights being the fraction of the class population that belongs to the corresponding modes. This baseline thus uses additional data (population statistics) that is not available for our approach. Also note that in the real-valued setup this weighted mean produces attribute descriptions that are exactly the same as those used in traditional ZSL benchmarks.
- **No aggregation.** This approach does not aggregate the provided modes. Instead, it simply treats all provided attribute vectors as equally applicable to all images in the class. Concretely, in the CADA-VAE framework, each iteration randomly samples *both images and attribute vectors* for each class. This has the effect of creating a latent embedding space where all the images of a class and the attribute vectors of all the modes would be pushed closer to each other.
- **Deployment only.** This approach uses *unimodal* attribute annotations during representation learning, but leverages multimodal annotations during deployment, where it operates exactly like MZSL.

All methods use CADA-VAE as the base learner. Table 5 compares our approach to these baselines. Our method clearly performs better than all the baselines on CUB and SUN. This is because it reasons about which images belong to which modes, thus making better use of the multimodal attributes. Also using multimodal attributes in deployment with a model trained on unimodal attributes performs worse than UZSL. This could be because, along with a poor representation there is a distribution shift when going from unimodal to multimodal descriptions. These results shows that only having multimodal data is not enough, we also need methods that can appropriately use such information.

In the real-valued setting, Mean of Modes performs worse than its unimodal counterpart (see Table 2), which is identical to Weighted Mean of Modes. One of the reasons why Weighted Mean of Modes works better than Mean of Modes in this setting is that it has additional information about the *frequency* of each mode. Since the train and test images are coming from the same distribution, biasing the learner with such population statistics is thus useful. However this bias might be detrimental if the population statistics used during training do not match that observed during deployment. Our method on the other hand requires no population statistics and hence is less likely to be perturbed by change in statistics when deployed.

5.3. Do we need mode labels for images?

In our multimodal benchmarks, the annotator provides us with multiple modes for each class, but the images in the base dataset are only annotated at the class level. One might consider a different alternative: to simply annotate the dataset at a much finer grain, by annotating images in the base dataset with the corresponding mode. While this requires a much more expensive annotation effort , it might lead to better representations and ultimately higher accuracy.

We compare our multimodal approach to this more expensive annotation strategy, which we call "Mode-annotated multimodal ZSL" or "Mode-annotated MZSL". Table 5 shows the performance of Mode Annotated MZSL in comparison to our approach as well as the multimodal ZSL baselines above. The base learner is CADA-VAE. On most of the datasets CUB, SUN (real-valued and binary) MZSL is able the achieve the maximum performance it can achieve *without requiring any mode annotations*. On DF real-valued, there is still a gap between MZSL and the upper-bound. Since DF has very few classes (25) and many attributes (1000), we speculate that our method cannot learn the optimal assignment for different modes and images leading to this gap. Nevertheless, these results suggest that mode annotations are in fact not necessary; we can get all the benefits of multimodal reasoning without these expensive annotations.

5.4. Ablations

We now evaluate the importance of various components of our approach. We first look at the learning schedule for MZSL. We compare the performance of our method MZSL on CUB, to MZSL without one of these phases. Table 6, shows the performance of these methods with CADA-VAE. Removing any of the three phases leads to a drop in performance, and thus all three phases are important. Removing the first phase leads to a very unstable mode assignment initially as the network is randomly initialized. Hence, phase 1 is extremely important for the stability of training. Phase 2 is the smooth transition phase from soft assignments to hard assignments. Hence, removing it also leads to a signifi-

	Real-valued Attributes			Binary Attributes		
Method	**CUB**	**SUN**	**DF**	**CUB**	**SUN**	**DF**
Mean of Modes	57.1 ± 0.5	55.5 ± 0.3	53.1 ± 0.3	42.2 ± 0.5	44.2 ± 0.3	$\mathbf{47.1 \pm 0.5}$
Weighted Mean of Modes	58.8 ± 0.2	59.4 ± 0.4	$\mathbf{58.0 \pm 0.3}$	42.4 ± 0.3	45.2 ± 0.5	$\mathbf{47.6 \pm 0.3}$
No aggregation	56.2 ± 0.3	55.6 ± 0.5	$\mathbf{57.5 \pm 0.5}$	38.9 ± 0.5	44.9 ± 0.5	46.4 ± 0.7
Deployment only	57.8 ± 0.3	58.9 ± 0.25	$\mathbf{57.9 \pm 0.3}$	40.4 ± 0.2	46.5 ± 0.28	$\mathbf{47.0 \pm 0.3}$
MZSL	$\mathbf{60.2 \pm 0.4}$	$\mathbf{62.0 \pm 0.3}$	$\mathbf{58.3 \pm 0.4}$	$\mathbf{44.0 \pm 0.3}$	$\mathbf{47.0 \pm 0.3}$	$\mathbf{47.4 \pm 0.7}$
Mode Annotated MZSL	60.5 ± 0.4	62.3 ± 0.2	59.4 ± 0.3	44.4 ± 0.3	47.2 ± 0.4	47.2 ± 0.9

Table 5. Comparison of MZSL with Mean ZSL and Weighted Mean ZSL on all datasets with real-valued and Binary setup and CADA-VAE as base model. Results show top-1 per class unseen accuracy averaged over 6 runs. Our method MZSL performs better than both these methods and is better at utilizing multimodal data.

Method	**Real-valued**	**Binary**
MZSL (no phase 1)	2.0 ± 0.0	2.0 ± 0.0
MZSL (no phase 2)	59.9 ± 0.3	40.1 ± 0.5
MZSL (no phase 3)	$\mathbf{60.1 \pm 0.3}$	42.2 ± 0.2
MZSL	$\mathbf{60.2 \pm 0.4}$	$\mathbf{44.0 \pm 0.3}$

Table 6. Performance of MZSL with one of the phase missing on CUB. Removing any step from training leads to a drop in model performance.

Method	**Real-valued**	**Binary**
MZSL (latent interpolation)	58.4 ± 0.5	39.5 ± 0.2
MZSL	$\mathbf{60.2 \pm 0.4}$	$\mathbf{44.0 \pm 0.3}$

Table 7. Performance of MZSL on CUB if we interpolate in the latent space of the model rather than attribute space.

cant drop in performance on both benchmarks. Removing phase 3 (hard assignments) has no impact on the real-valued benchmark but reduces accuracy on the binary benchmark.

We next look at another variant of our method, where instead of interpolating attributes in the attribute space we interpolate it in latent space, as discussed in Sec. 3.2. Table 7, shows the performance of this variant with CADA-VAE. Interpolating in latent space is not very useful and leads to a very significant drop in the performance. This suggests that the semantics of the learned latent space is substantially different from that of the attribute space.

5.5. Qualitative Results

Figure 4, shows t-SNE [25] visualization of 2 CUB classes in the representation space CADA-VAE trained with UZSL (left) and MZSL (right). For both "Rusty Blackbird" and "Cardinal" we see that in the latent embedding of UZSL images of all different modes are closer together and not clearly separable. Whereas, with MZSL we can clearly see separated modes. For "Cardinal" the cluster in the top right are females whereas the big cluster in the center are males. For "Rusty Blackbird" the cluster of images in the top-left are breeding males, clustered to the right are non-breeding males and the small cluster at the center are females.

MZSL creates a better representation where visually different looking birds of the same species are not all closer together. This results in a better generalization to unseen

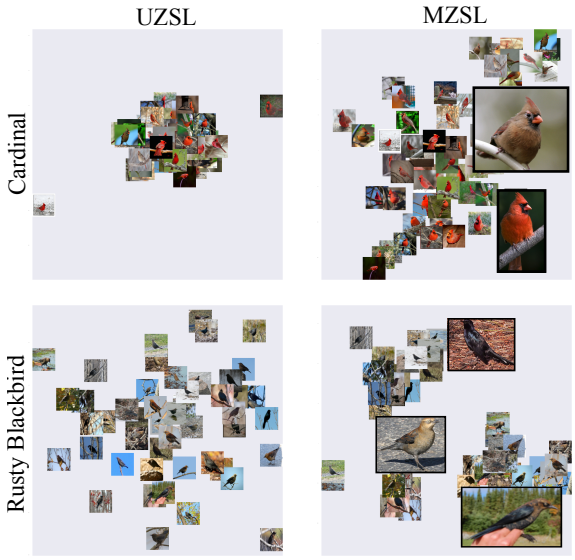

Figure 4. t-SNE visualization of CUB classes in the latent embedding space of UZSL vs MZSL in the binary attribute setting. Bigger images with black border show one of the images belonging to the mode. For both classes, MZSL creates encodings that look separable by different looking modes, while UZSL forces the modes closer together (Zoom in to see more details).

classes and hence the better accuracies. See supplementary for more examples.

6. Conclusion

In this work we have shown that single attribute vectors are not sufficient to capture class appearance, negating a key assumption in prior zero-shot learning work. As an alternative, we have proposed the problem of multimodal zero-shot learning, where the annotator specifies an attribute vector for each *mode* of appearance. We have presented a new benchmark (both automatic and manual) for this problem and as a solution to this task, we have presented a multimodal technique that gains up to **5 points** in accuracy compared to their unimodal counterparts. We show that our technique is generalizable to many existing (and possibly future) zero-shot learning models as long as they have a common embedding space for images and attributes.

References

[1] Zeynep Akata, Scott Reed, Daniel Walter, Honglak Lee, and Bernt Schiele. Evaluation of output embeddings for fine-grained image classification. In *CVPR*, 2015. 1, 2, 3

[2] Soravit Changpinyo, Wei-Lun Chao, Boqing Gong, and Fei Sha. Synthesized classifiers for zero-shot learning. In *CVPR*, 2016. 1, 2

[3] Mohamed Elhoseiny, Yizhe Zhu, Han Zhang, and Ahmed Elgammal. Link the head to the" beak": Zero shot learning from noisy text description at part precision. In *CVPR*, 2017. 2

[4] Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. Describing objects by their attributes. In *CVPR*, 2009. 2

[5] Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. In *NeurIPS*, 2013. 2

[6] Elyor Kodirov, Tao Xiang, Zhenyong Fu, and Shaogang Gong. Unsupervised domain adaptation for zero-shot learning. In *CVPR*, 2015. 1, 2, 3

[7] Elyor Kodirov, Tao Xiang, and Shaogang Gong. Semantic autoencoder for zero-shot learning. In *CVPR*, 2017. 1, 2, 3

[8] Vinay Kumar Verma, Gundeep Arora, Ashish Mishra, and Piyush Rai. Generalized zero-shot learning via synthesized examples. In *CVPR*, 2018. 2

[9] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*. IEEE, 2009. 2

[10] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Attribute-based classification for zero-shot visual object categorization. *TPAMI*, 2013. 2

[11] Hugo Larochelle, Dumitru Erhan, and Yoshua Bengio. Zero-data learning of new tasks. In *AAAI*, 2008. 2

[12] Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther. Autoencoding beyond pixels using a learned similarity metric. In *ICML*, 2016. 5

[13] Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *CVPR*, 2016. 2, 5

[14] Utkarsh Mall, Bharath Hariharan, and Kavita Bala. Field-guide-inspired zero-shot learning. In *ICCV*, 2021. 2

[15] Ashish Mishra, Shiva Krishna Reddy, Anurag Mittal, and Hema A Murthy. A generative model for zero shot learning using conditional variational autoencoders. In *CVPR Workshops*, 2018. 2

[16] Sanath Narayan, Akshita Gupta, Fahad Shahbaz Khan, Cees GM Snoek, and Ling Shao. Latent embedding feedback and discriminative features for zero-shot classification. *ECCV*, 2020. 1, 2, 3, 5

[17] 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 *ICLR*, 2013. 2

[18] Genevieve Patterson and James Hays. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *CVPR*, 2012. 2, 5

[19] Tzuf Paz-Argaman, Reut Tsarfaty, Gal Chechik, and Yuval Atzmon. Zest: Zero-shot learning from text descriptions using textual similarity and visual summarization. In *EMNLP: Findings*, 2020. 2

[20] Ruizhi Qiao, Lingqiao Liu, Chunhua Shen, and Anton Van Den Hengel. Less is more: zero-shot learning from online textual documents with noise suppression. In *CVPR*, 2016. 2

[21] Bernardino Romera-Paredes and Philip Torr. An embarrassingly simple approach to zero-shot learning. In *ICML*, 2015. 1, 2, 3

[22] Edgar Schonfeld, Sayna Ebrahimi, Samarth Sinha, Trevor Darrell, and Zeynep Akata. Generalized zero-and few-shot learning via aligned variational autoencoders. In *CVPR*, 2019. 1, 2, 3, 4, 5

[23] Yutaro Shigeto, Ikumi Suzuki, Kazuo Hara, Masashi Shimbo, and Yuji Matsumoto. Ridge regression, hubness, and zero-shot learning. In *ECML PKDD*. 1, 2

[24] Richard Socher, Milind Ganjoo, Christopher D Manning, and Andrew Ng. Zero-shot learning through cross-modal transfer. In *NeurIPS*, 2013. 2

[25] Laurens van der Maaten and Geoffrey E. Hinton. Visualizing high-dimensional data using t-sne. *JMLR*, 9, 2008. 8

[26] Vinay Kumar Verma and Piyush Rai. A simple exponential family framework for zero-shot learning. In *ECML*, 2017. 1, 2

[27] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011. 2, 5

[28] Yongqin Xian, Zeynep Akata, Gaurav Sharma, Quynh Nguyen, Matthias Hein, and Bernt Schiele. Latent embeddings for zero-shot classification. In *CVPR*, 2016. 2

[29] Yongqin Xian, Christoph H Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly. *TPAMI*, 2018. 2, 5, 6

[30] Yongqin Xian, Tobias Lorenz, Bernt Schiele, and Zeynep Akata. Feature generating networks for zero-shot learning. In *CVPR*, 2018. 2, 3

[31] Yongqin Xian, Saurabh Sharma, Bernt Schiele, and Zeynep Akata. f-vaegan-d2: A feature generating framework for any-shot learning. In *CVPR*, 2019. 1, 2

[32] Yizhe Zhu, Mohamed Elhoseiny, Bingchen Liu, Xi Peng, and Ahmed Elgammal. A generative adversarial approach for zero-shot learning from noisy texts. In *CVPR*, 2018. 2