Supplementary Material for "Zero-shot Learning Using Multimodal Descriptions"

In the following supplementary material we include additional information and experiments that we could not present in the main paper. In section 1 we present how we created the multimodal benchmark for evaluating our methods. In section 2 we show the interface for collecting manual mode annotations on CUB. In section 3 we present MZSL performance when using TF-VAEGAN on the SUN and DF datasets. Finally, section 4 shows more examples of mode separation when using MZSL instead of UZSL.

1. 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 for 3 datasets CUB, SUN and DeepFashion.

1.1. Unimodal ZSL Benchmarks

Existing Unimodal ZSL benchmarks for CUB, SUN and DeepFashion are created using image-level attribute annotations. For a category c, let $\{\mathbf{x}_{c_1}, \mathbf{x}_{c_2} \cdots \mathbf{x}_{c_m}\}$ be the the images belonging to the class. The annotators are shown images and asked for attribute annotation. This process results in image-level attribute annotations, as for each each image $\mathbf{x}_{c_k}$ one gets attribute annotations α_{c_k}. In unimodal ZSL benchmarks, the image-level attributes are aggregated for each class to get class-level attributes. To get a real-valued class attribute description one can take the average of image-level attribute annotation of that class. So the real-valued class attribute for class c would be

$$\mathbf{a}(c) = \mu(\{\alpha_{c_1}, \alpha_{c_2} \cdots \alpha_{c_m}\}) \tag{1}$$

Since the annotations are marked as 0 or 1, to show the absence or presence of a particular attribute in an image, to get a binary class attribute description one can take the median value of image-level attribute annotation of that class. So the binary class attribute for class c would be

$$\mathbf{a}(c) = \text{median}(\{\alpha_{c_1}, \alpha_{c_2} \cdots \alpha_{c_m}\}) \tag{2}$$

Note that this way of creating benchmark for zero-shot learning would be impractical in real-world application. In order to get attribute descriptions for an unseen class, we would need to collect labeled images for that class which breaks the zero-shot assumptions. On top of that annotators would need to annotate multiple attributes for all these images which would be expensive. Nonetheless, for the purpose of benchmarking zero-shot methods this produces, very high-quality attributes and is thus used in practice. Hence we also use similar pipeline to create multimodal benchmarks.

1.2. Multimodal ZSL Benchmarks

To create a multimodal ZSL benchmark out of image-level attribute data, we need to *cluster* image-level attribute annotations in each class into a set of modes. We considered two ways of performing such clustering: an automatic approach, and an approach involving manual annotation.

In the automatic approach, we cluster per-image attribute annotations, by using a variant of k-means [2], called k-pod[1], that can handle missing values while clustering. K-pod keeps running in three phases iteratively until convergence, 1) Filling the missing value 2) Classify points to nearest mean 2) Update the mean values. The last two steps are same as that in k-means. In the first step, the missing values are replaced with mean of values from the same cluster that are not missing. A key question is also how many clusters to choose for each class. We find the optimal number of modes using the silhouette score [3]. The silhouette score is a metric used to measure the goodness of clustering with respect to the number of modes. It uses intra-cluster and inter-cluster distances to measure goodness.

A potential concern is whether the automatic approach actually identifies the right modes. Therefore, we also designed a manual approach, where the clusters from the automatic approach are validated and corrected by human annotators. In the first step annotators are asked to remove outlier images from these noisy clusters. In the second step, annotators are asked to merge cleaned clusters and place outliers in either a new or existing cluster. Both the number of clusters and images in clusters are thus decided by the human annotators. Refer to section 2 for more information about this interface for manual clustering.

Once we obtain the different clusters and the images in that cluster, the multimodal attributes are obtained by aggregating the image-level attributes of that cluster. This step is similar to aggregation described in eq. (1) and (2).

As stated in main paper, we tested both the automatic and manual approach on CUB, creating two multimodal ZSL benchmarks for CUB. We observed very similar results on both benchmarks (see Table 1 (main paper)). As such, for SUN and DeepFashion, we only used the automatic approach to create the multimodal benchmark.

2. Interface

Our interface for collecting manual labels consist of two steps. As discussed in the main paper, we first find out clusters using the automatic method, and annotators use those as an initialization for further cleaning. In the first step annotators are asked to remove outlier images from these noisy clusters. In the second step, annotators are asked to merge cleaned clusters and place outliers in either a new or existing cluster. Both the number of clusters and images in clusters are thus decided by the human annotators.

Figure 1 shows an example of the interface presented to the annotator in the first step, for one of the automated clusters of a bird species "Mallard". The annotators are asked to clean it, by selecting images and removing them from the clusters. The annotators can also undo their decisions and add back an image to the original cluster. Note the the annotators can also click the link for the bird at the top that would redirect them to an ornithology website about the bird[1], that could be used to learn about the modes of the bird category. Images removed in this step can be added later to one of the manually found modes in the next step.

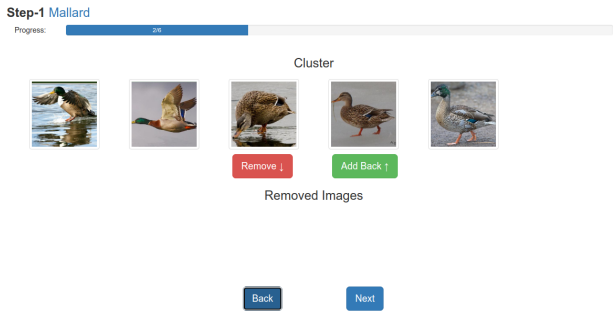

Figure 1. First step of our annotation interface where the annotators clean the automatically found clusters.

Figure 2 shows an example of the interface presented to the annotator in the second step, for one of the automated clusters of a bird species "Pine Grosbeak". The annotators are asked to merge similar looking clusters and put the outlier images found in the previous step in one of the existing clusters or create a new cluster. Additionally, annotators can further clean a cluster by removing images from them.

[1] https://www.allaboutbirds.org/guide/Mallard

3. TF-VAEGAN performance on SUN and DF

Table 1 shows the performance of MZSL using TF-VAEGAN on the SUN and DF datasets. MZSL is significantly better than using unimodal attributes. Table 1 also shows the performance of MZSL with TF-VAEGAN on SUN and DF in comparison to other baselines. While on DF the gains by using multimodal attributes are not significant, on SUN this is not the case. In the binary setting using multimodal attributes and our model leads to a 6.8% gain as compared to when using the unimodal dataset.

4. More Qualitative Results

We look at more t-SNE visualizations of bird classes. Figure 3 and 4 show t-SNE visualization for 6 more classes of birds where MZSL results in clear mode separation and subsequently better recognition. We show the classes where clear mode separation is visible for MZSL.

References

[1] Jocelyn T Chi, Eric C Chi, and Richard G Baraniuk. k-pod: A method for k-means clustering of missing data. *The American Statistician*, 2016. 1

[2] James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *BSMSP*, 1967. 1

[3] Peter J Rousseeuw. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *JCAM*, 1987. 1

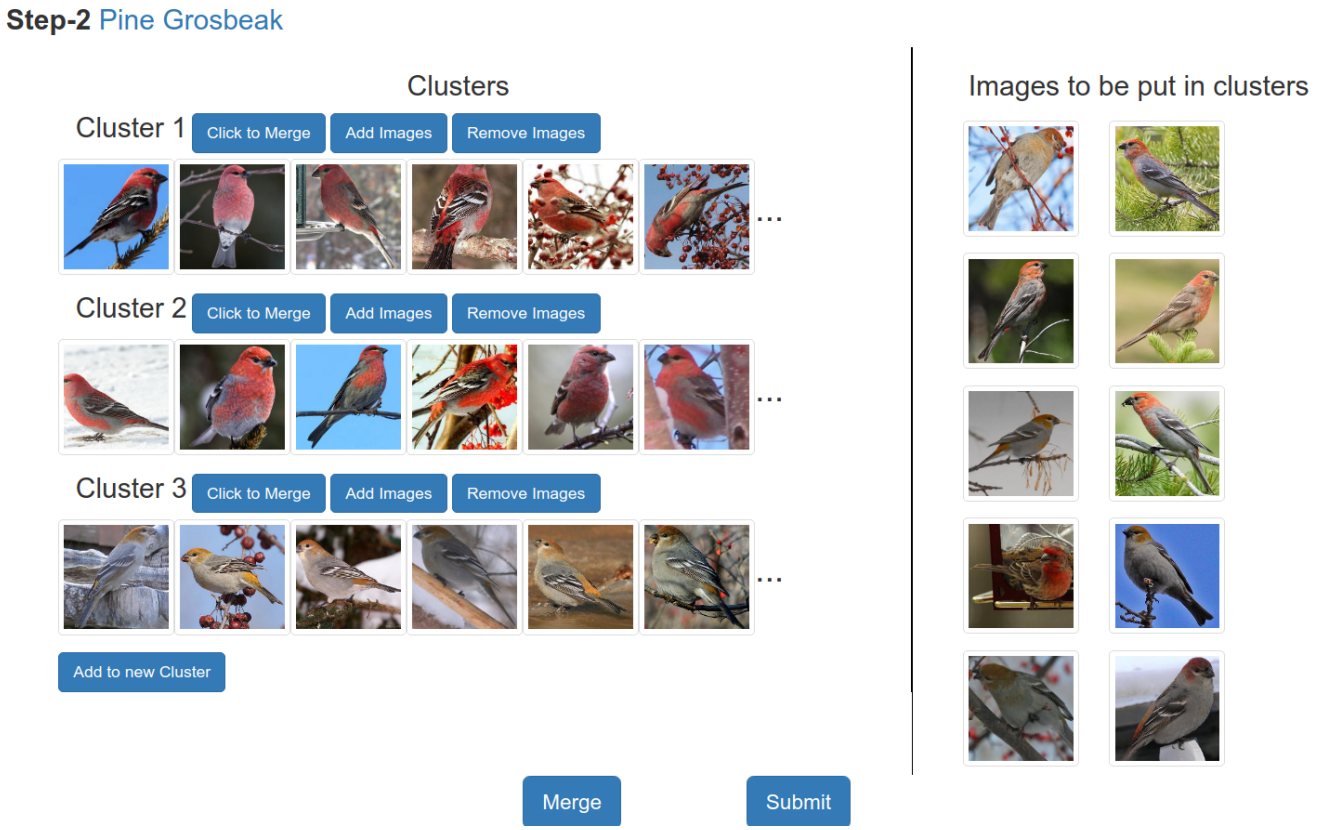

Figure 2. Second step of our annotation interface where the annotators merge and create new clusters and add the outlier images to one of the clusters found in step 1.

	TF-VAEGAN			
	SUN		DF	
	Real-valued	Binary	Real-valued	Binary
UZSL	$\mathbf{65.7 \pm 0.4}$	44.4 ± 0.4	$\mathbf{59.3 \pm 0.5}$	47.1 ± 0.6
MZSL	$\mathbf{65.9 \pm 0.6}$	$\mathbf{51.2 \pm 0.7}$	$\mathbf{59.6 \pm 0.4}$	$\mathbf{48.2 \pm 0.4}$
Mean of Modes	62.9 ± 1.0	46.9 ± 2.0	58.4 ± 0.2	$\mathbf{47.6 \pm 0.5}$
Weighted Mean of Modes	$\mathbf{65.7 \pm 0.4}$	50.9 ± 71	$\mathbf{59.3 \pm 0.5}$	$\mathbf{48.1 \pm 0.4}$
Mode Annotated MZSL	66.2 ± 0.44	51.4 ± 0.1	59.5 ± 0.2	48.1 ± 0.9

Table 1. Comparison of UZSL (and other baselines) with MZSL on SUN and DF dataset with **TF-VAEGAN** as the base learner, with real-valued and binary attributes.

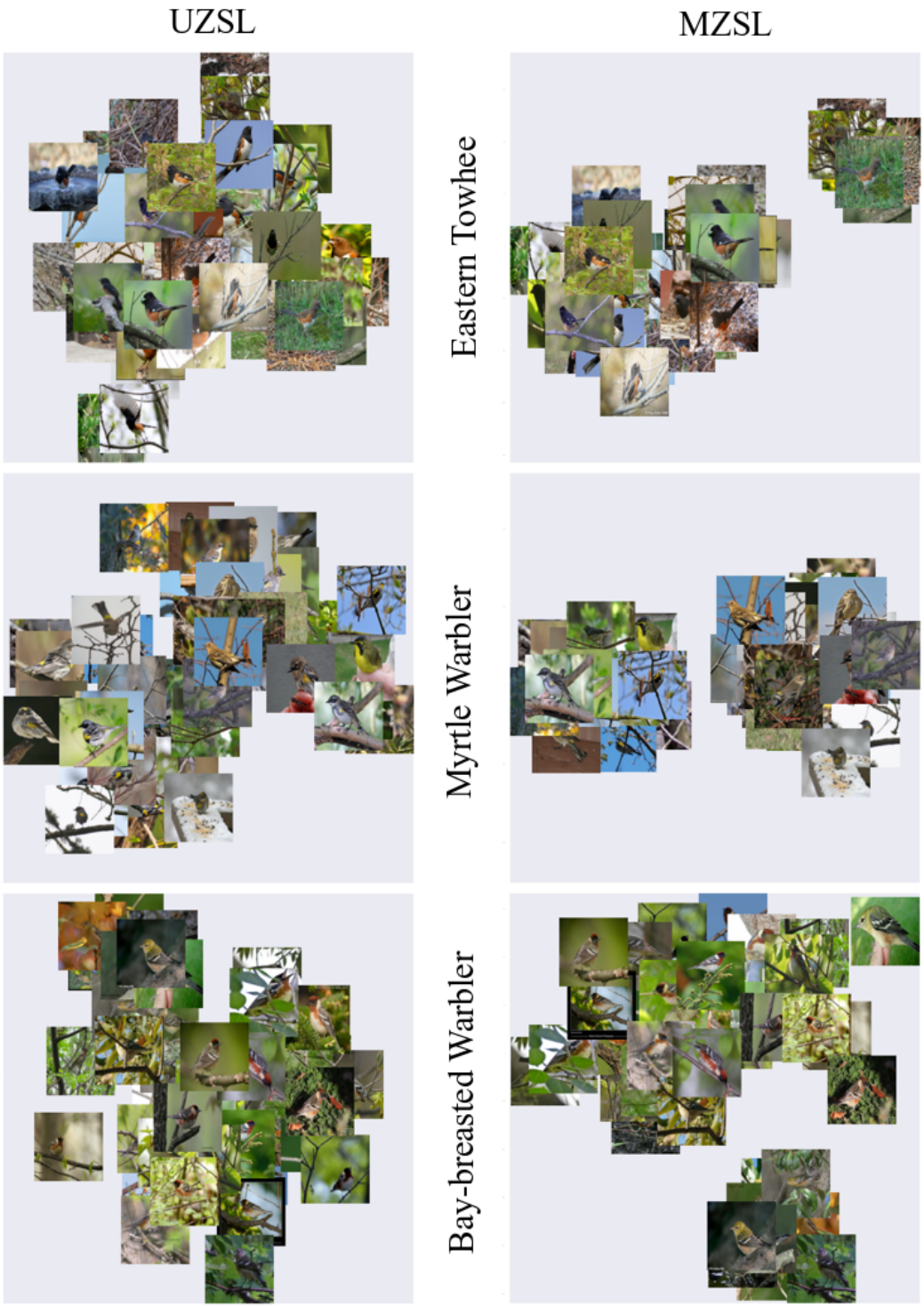

Figure 3. t-SNE visualization of classes from CUB with UZSL (left) and MZSL (right). Clear mode separation can be seen in the case of MZSL.

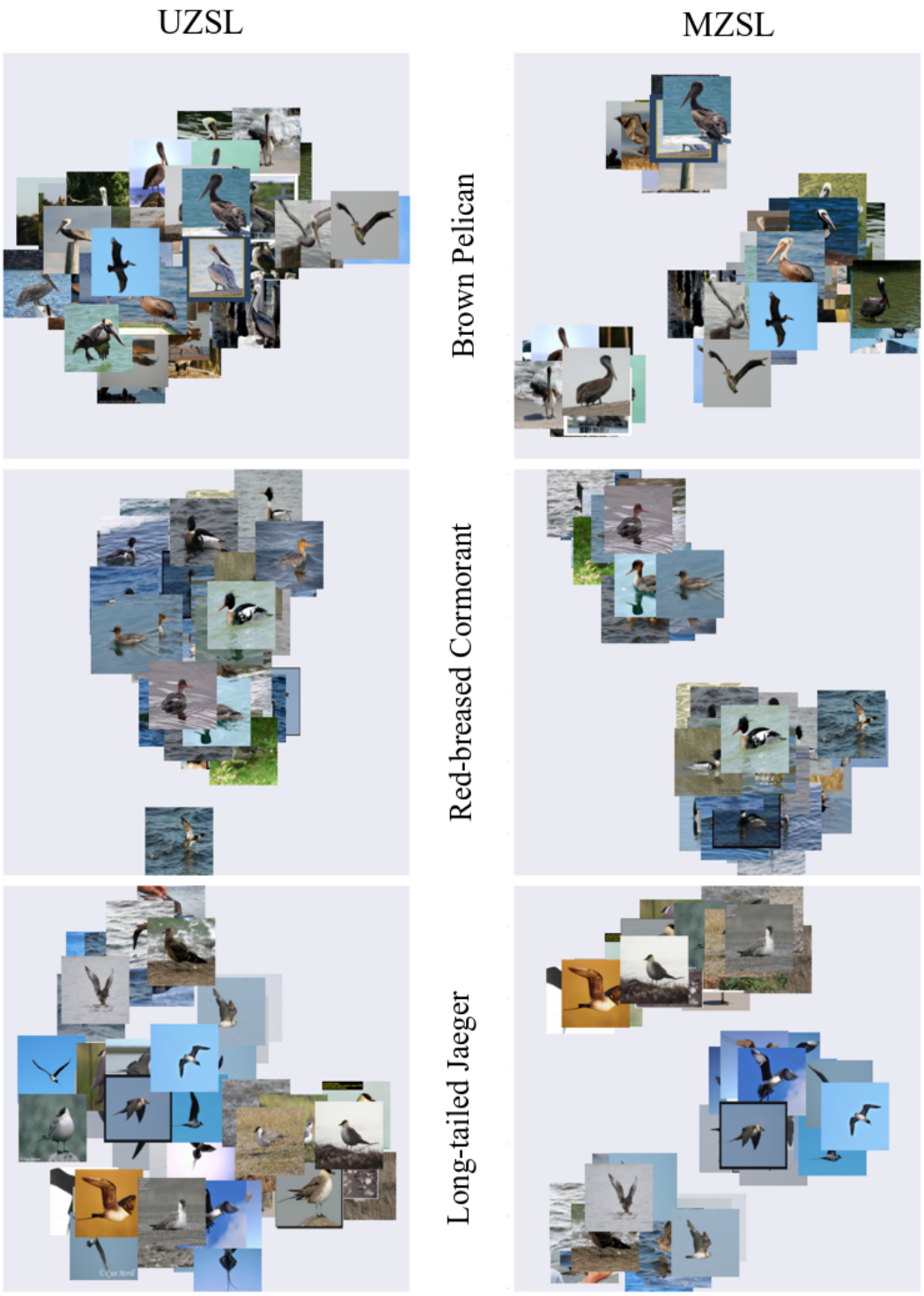

Figure 4. More examples of t-SNE visualization of classes from CUB with UZSL (left) and MZSL (right).