Supplementary Materials of Region Graph Embedding Network for Zero-Shot Learning

Anonymous ECCV submission

Paper ID 2495

1 Other Approaches for Calculating Contrasting Class Similarity

Suppose the l_2-normalized attribute matrix w.r.t. the C^s seen classes and C^u unseen classes are $A \in \mathbb{R}^{Q \times C^s}$ and $B \in \mathbb{R}^{Q \times C^u}$, respectively. We have calculated the contrasting class similarity $V \in \mathbb{R}^{C^u \times C^s}$ using least square regression (LSR) in Line 324 (§3.4) of the manuscript. We further assess the scalability of RGEN towards other types of calculating for V. Specifically, we take CUB [1] and AWA2 [2] as example datasets and compare the following three types of calculations for V:

$$\texttt{LSR:} \qquad V = (B^\intercal B + \beta I)^{-1} B^\intercal A. \tag{1}$$

$$\texttt{Cosine:} \qquad V = B^\intercal A. \tag{2}$$

$$\texttt{Exponential Cosine:}\ V = \exp(B^\intercal A). \tag{3}$$

Note that, the dot-product in Eq. (2) and (3) equals to cosine similarity metric, since each column of A and B is l_2 normalized.

Table 1: Comparisons of RGEN performances (%) under ZSL and GZSL w.r.t. three types of calculations for V.

Types	**CUB**				**AWA2**			
	ZSL	GZSL			ZSL	GZSL		
	MCA	ts	tr	H	MCA	ts	tr	H
LSR	**76.1**	60.0	73.5	**66.1**	**73.6**	67.0	76.5	71.5
Cosine	75.2	58.3	71.8	64.3	73.0	67.3	77.1	**71.8**
Exp Cosine	74.9	58.7	71.8	64.6	72.5	67.3	76.3	71.5

Both the ZSL and GZSL [2] results are shown in Table 1. It can be seen that 1) LSR consistently achieves a better MCA (mean class accuracy) under ZSL on both CUB and AWA2, and 2) LSR achieves a better H under GZSL on CUB, and meanwhile, a slightly better H is obtained by Cosine metric under GZSL on AWA2. All three methods perform well with a desirable performance. This validates the scalability of the proposed RGEN.

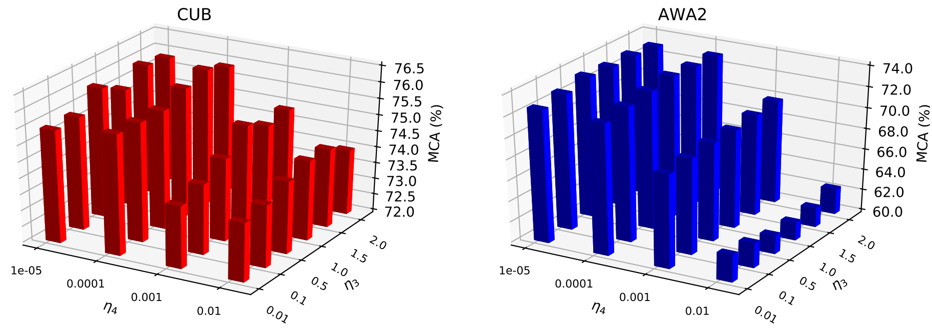


Fig. 1: MCA-(η_3, η_4) maps of RGEN under ZSL.

2 Effects of Coefficients η_3 and η_4 to RGEN

η_3 and η_4 are the trade-off parameters w.r.t. the compact loss $\mathcal{L}_{\text{cpt}}$ and the divergent loss $\mathcal{L}_{\text{div}}$, respectively. For RGEN training (Eq. (11) of §3.5 in the manuscript), we have fixed η_3 and η_4 to 1.0 and 0.0001 on all the four datasets used, respectively. By further taking the values of η_3 from $\{0.01, 0.1, 0.5, 1.0, 1.5, 2.0\}$ and the values of η_4 from $\{1e-5, 1e-4, 1e-3, 1e-2\}$, we observe the MCA of RGEN w.r.t. different combinations of (η_3,η_4) for ZSL, on CUB and AWA2 datasets (Fig. 1). We find that a small η_4, meanwhile, a relative large η_3 are better for assisting the RGEN model. As such, we set $(\eta_3, \eta_4) = (1.0, 0.0001)$ for all datasets.

3 More t-SNEs of the Features in the Semantic Space

We have only illustrated the t-SNEs [3] (on unseen test images of AWA2) of RGEN and its variants (CPA and PRR) under GZSL, due to space limitation. We further take CUB and AWA2 as examples to visualize the feature representations of both seen and unseen test images in the semantic space for RGEN, CPA and PRR. Fig. 2 and Fig. 3 illustrate the t-SNEs of RGEN, CPA and PRR for AWA2 and CUB, respectively.

4 More Qualitative Analysis on Attended Parts

In our manuscript, we have used unseen images from CUB under ZSL to visualize the attended parts (Fig. 8 of the manuscript). Compared with the baseline (which achieves a 71.3% MCA when trained by only the ACE loss, Table 4 of the manuscript), RGEN has shown some useful insights, e.g., it can 1) discover more divergent parts w.r.t. objects; 2) suppress background and redundant foreground regions (maximum mask values in parts #1-4, 9-10 are all small and no similar masks exist among foreground parts #5-8); and 3) automatically align the order relationships of different parts (parts #5-8 are consistent w.r.t. different unseen

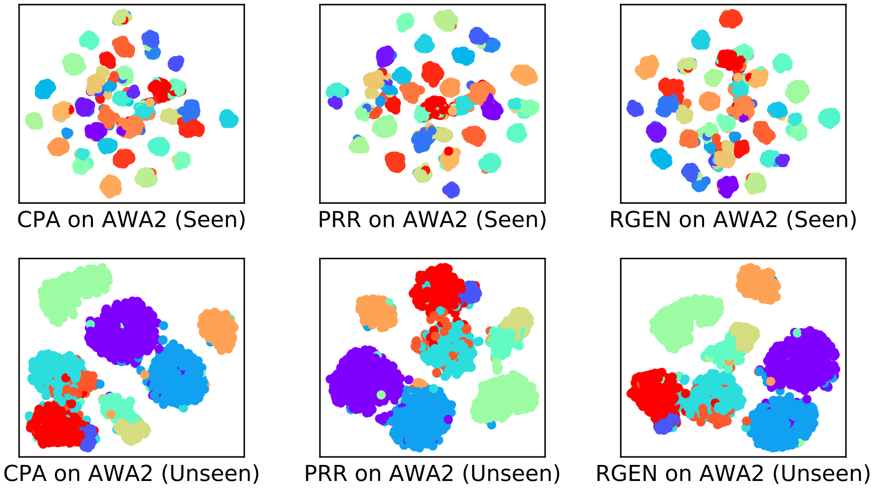


Fig. 2: t-SNEs of the projected features in semantic space on AWA2.

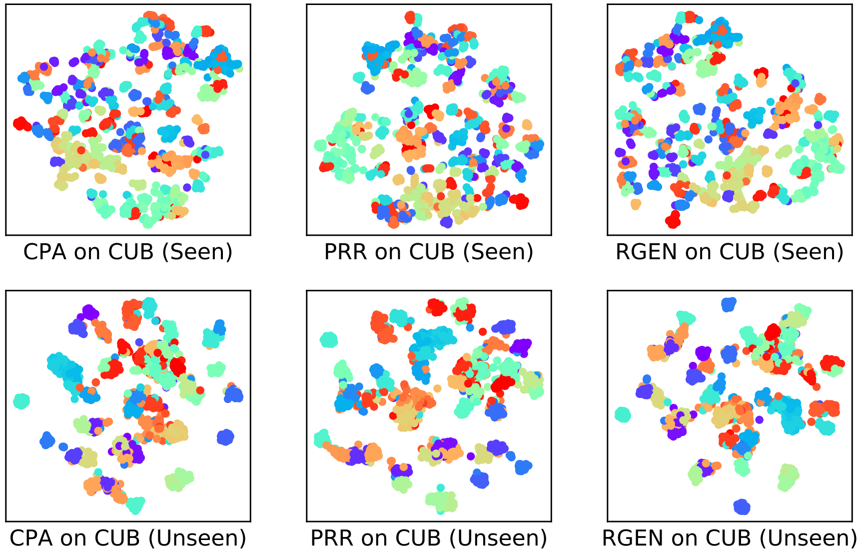


Fig. 3: t-SNEs of the projected features in semantic space on CUB.

class images). Here, we take the same trained model as used for drawing Fig. 8 of the manuscript and give more visualizations in Fig. 4 and Fig. 5 on CUB dataset. Note that the showed images are randomly selected from the unseen test set without any human intervention. Fig 4 and Fig. 5 again show the same qualitative conclusions as previously mentioned. This means that our RGEN with parts relation reasoning has discovered some intrinsic reasons for unseen class recognition, e.g., the model can automatically align the order relationships of different parts.

Furthermore, we illustrate the attended parts of both seen and unseen images on CUB, under the best RGEN GZSL model (Fig. 6 and Fig. 7). Note that the showed images are also randomly selected from the unseen test set and seen test set without any human intervention.

5 More Comparisons of Prediction Results of Unseen Test Images under GZSL

As the domain bias issue is a typical problem under GZSL, we have shown an example in Fig. 3 of our manuscript, by feeding two test unseen images to the Baseline/RGEN GZSL models on AWA2. Here, we show more randomly selected examples for comparing the performances of Baseline (without *balance* loss) with our RGEN GZSL model, on AWA2 (Fig 8 and Fig. 9). In most cases, our RGEN can well address the domain bias issue encountered in the Baseline model.

6 Detailed Parameter Values for Each Dataset

As stated in §4.2 and §3.5 of the manuscript, we totally have eight key parameters: η_1, η_2, η_3, η_4, λ_1, λ_2, K, and GCN Architecture in §3.3. We further illustrate their taken values to achieve the results in Tables for each dataset (Table 2). It can be seen that six out of eight parameters are fixed for all used datasets, therefore, only λ_1, λ_2 are parameters that need to be tuned. However, as can be seen from §4.5 of the manuscript, these two parameters are also robust to the final MCA and H score. This indicates that the RGEN model is essentially a scalable model to tackle ZSL and GZSL tasks.

7 Component Analysis w.r.t. tr, ts and H under the best RGEN GZSL model

We have conducted component analysis w.r.t. H for GZSL in Table 5 of §4.5 in the manuscript, due to space limitation. In Table 3, we show all the results including tr, ts and H score for the same setting of componet analysis in the manuscript.

We conclude from Table 3 that **1)** our *balance* loss contributes mostly to the performance improvements of ts and H score; **2)** in some cases, the tr is also improved, e.g., all tr, ts and H are improved significantly on CUB; and **3)** sometimes, the tr is dropped to some extent; however, compared with such tolerable performance degradations on tr (e.g., 33.6%→31.0% on SUN and 52.4%→49.2% on APY), the improvements on ts and H are significant on all the used datasets.

Table 2: Detailed parameter values for each dataset.

Dataset	η_1	η_2	η_3	η_4	λ_1 for CPA	λ_2 for CPA	λ_1 for PRR	λ_2 for PRR	K	GCN
CUB	0.9	0.1	1.0	1e-4	0.05	0.05	0.05	0.05	10	2048-1024-2048
AWA2	0.9	0.1	1.0	1e-4	0.001	0.05	0.001	0.05	10	2048-1024-2048
SUN	0.9	0.1	1.0	1e-4	0.07	0.1	0.07	0.1	10	2048-1024-2048
APY	0.9	0.1	1.0	1e-4	0.01	0.07	0.01	0.07	10	2048-1024-2048

Table 3: Component analysis w.r.t. tr, ts and H under best GZSL RGEN model.

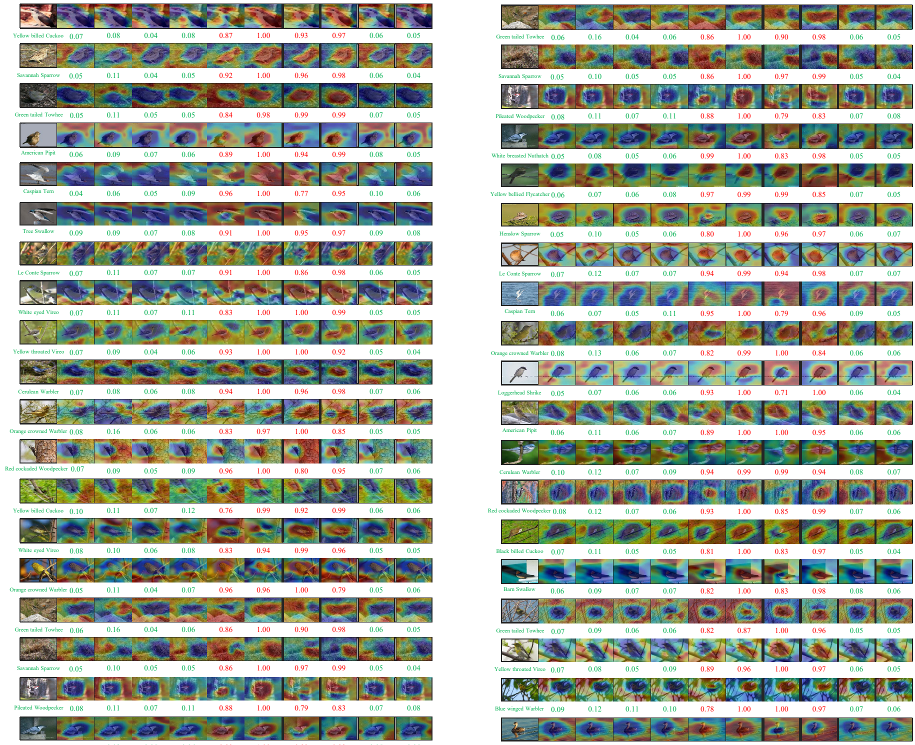
Transfer Loss	CD Regularization	PRR Branch	Balance Loss	CUB			AWA2			SUN			APY		
				ts	**tr**	**H**	**ts**	**tr**	**H**	**ts**	**tr**	**H**	**ts**	**tr**	**H**
✔				25.8	67.1	37.2	6.7	93.5	12.5	15.5	33.6	21.2	8.9	52.4	15.2
✔	✔			24.8	59.8	38.6	7.6	92.4	14.1	17.8	35.5	23.7	9.2	52.4	15.6
✔	✔	✔		28.0	67.3	39.6	8.0	92.6	14.7	18.7	34.9	24.3	9.6	56.0	16.4
✔		✔		26.7	67.5	38.3	8.1	92.1	14.9	17.8	34.9	23.6	9.2	56.9	15.8
✔			✔	61.7	67.8	64.6	66.8	73.3	69.9	42.8	31.0	35.9	29.5	49.2	36.8
✔	✔		✔	61.4	68.5	64.7	64.1	76.4	69.7	44.4	30.8	36.4	29.2	48.0	36.3
✔	✔	✔	✔	60.0	73.5	66.1	67.0	76.5	71.5	44.0	31.7	36.8	30.4	48.1	37.2
✔		✔	✔	62.3	68.2	65.1	67.1	75.9	71.3	44.2	31.4	36.7	30.4	49.5	37.7

Note that, in the real-world application, we want to correctly classify both seen and unseen test images as many as possible (i.e., we pursue a higher H score), but most of the recently proposed deep GZSL models [4,5] fail to achieve a balanced tr and ts, especially, the ts is usually very low (e.g., 26.4% for LDF and 36.2% for LFGAA on CUB). By contrary, our RGEN model has achieved a satisfactory H score (balanced tr and ts) in all used datasets. As such, the benefits brought by our model are much greater than the performance degradation of tr, which shows its potential to the real-world application.

References

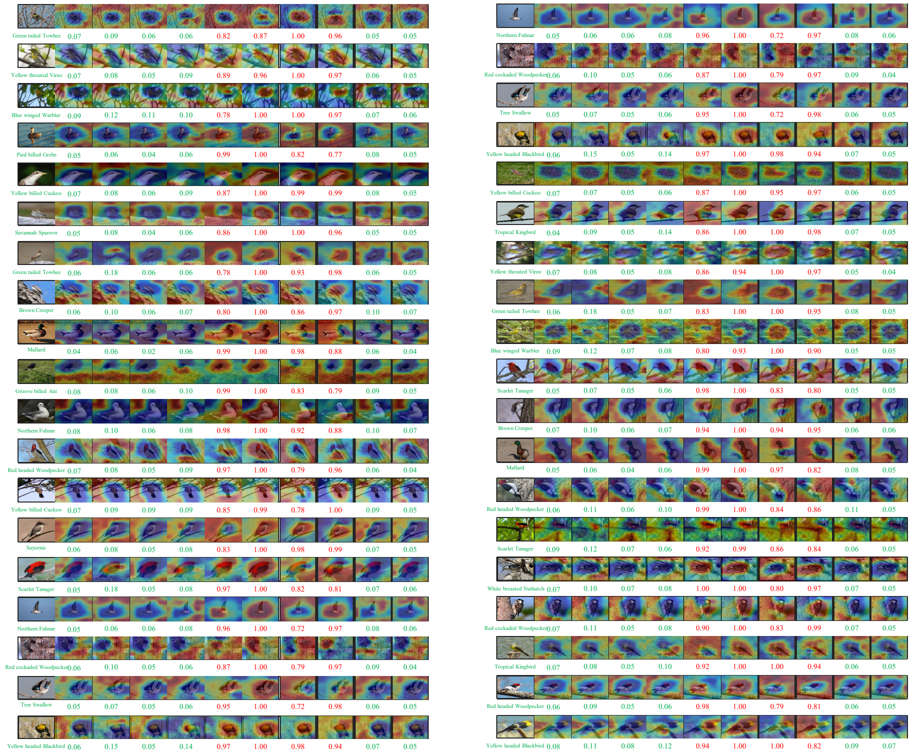
1. C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. In *Technical report*, 2011. 1
2. Y. Xian, B. Schiele, and Z. Akata. Zero-shot learning-the good, the bad and the ugly. In *arXiv:1703.04394*, 2017. 1
3. L. Maaten and G. Hinton. Visualizing data using t-SNE. In *JMLR*, 2008. 2
4. Y. Li, J. Zhang, J. Zhang, and K. Huang. Discriminative learning of latent features for zero-shot recognition. In *CVPR*, 2018. 5
5. Y. Liu, J. Guo, D. Cai and X. He. Attribute Attention for Semantic Disambiguation in Zero-Shot Learning. In *ICCV*, 2019.

5



(a) The 19 Randomly Selected Unseen Images for the 1*th* Time (b) The 19 Randomly Selected Unseen Images for the 2*th* Time

Fig. 4: More visualizations of the attended parts for unseen test images on CUB, under ZSL. For each row, the first one is the input image, the left ones are its ten attended parts, the numbers are the maximum value within corresponding mask (parts marked with green/red number are background and foreground parts, respectively). As concluded in our manuscript, RGEN can 1) discover more divergent parts w.r.t. objects; 2) suppress background and redundant foreground regions (maximum mask values in parts #1-4, 9-10 are all small and no similar masks exist among foreground parts #5-8); and 3) automatically align the order relationships of different parts (parts #5-8 are consistent w.r.t. different unseen class images). We randomly select four times from the unseen test image set: (a) The 19 Randomly Selected Unseen Images for the $1th$ Time. (b) The 19 Randomly Selected Unseen Images for the $2th$ Time. Zoom in four times to see details.



(c) The 19 Randomly Selected Unseen Images for the 3*th* Time

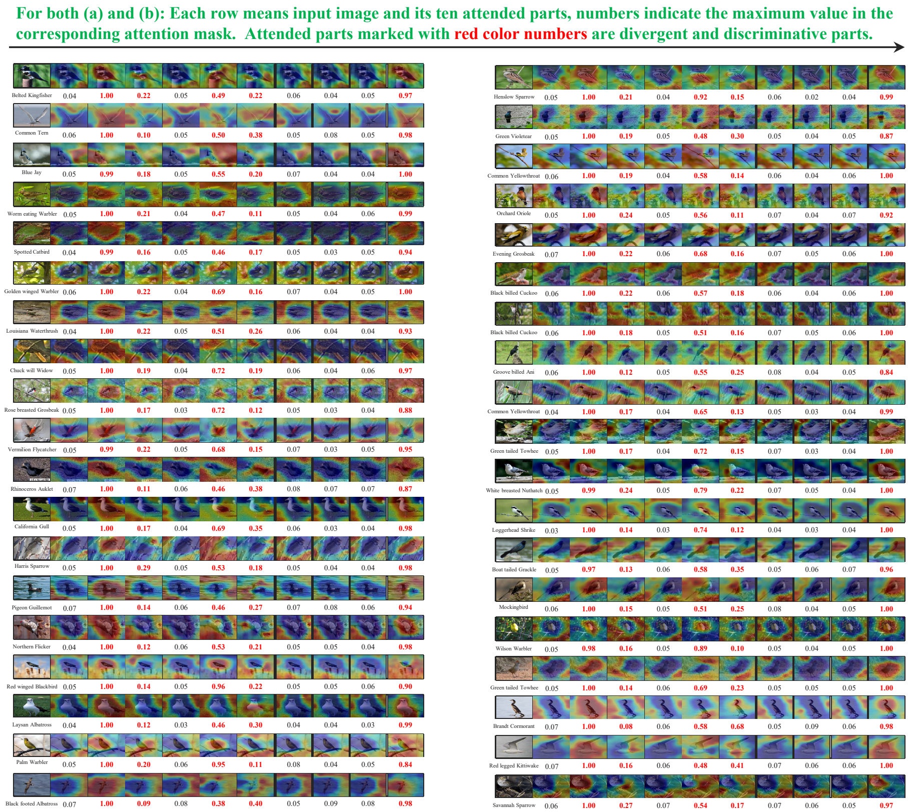
(d) The 19 Randomly Selected Unseen Images for the 4*th* Time

Fig. 5: More visualizations of the attended parts for unseen test images on CUB, under ZSL. For each row, the first one is the input image, the left ones are its ten attended parts, the numbers are the maximum value within corresponding mask (parts marked with green/red number are background and foreground parts, respectively). As concluded in our manuscript, RGEN can 1) discover more divergent parts w.r.t. objects; 2) suppress background and redundant foreground regions (maximum mask values in parts #1-4, 9-10 are all small and no similar masks exist among foreground parts #5-8); and 3) automatically align the order relationships of different parts (parts #5-8 are consistent w.r.t. different unseen class images). We randomly select four times from the unseen test image set: (c) The 19 Randomly Selected Unseen Images for the 3*th* Time. (d) The 19 Randomly Selected Unseen Images for the 4*th* Time. Zoom in four times to see details.

(a) GZSL: The 19 Randomly Selected Test Seen Images for the 1*th* Time

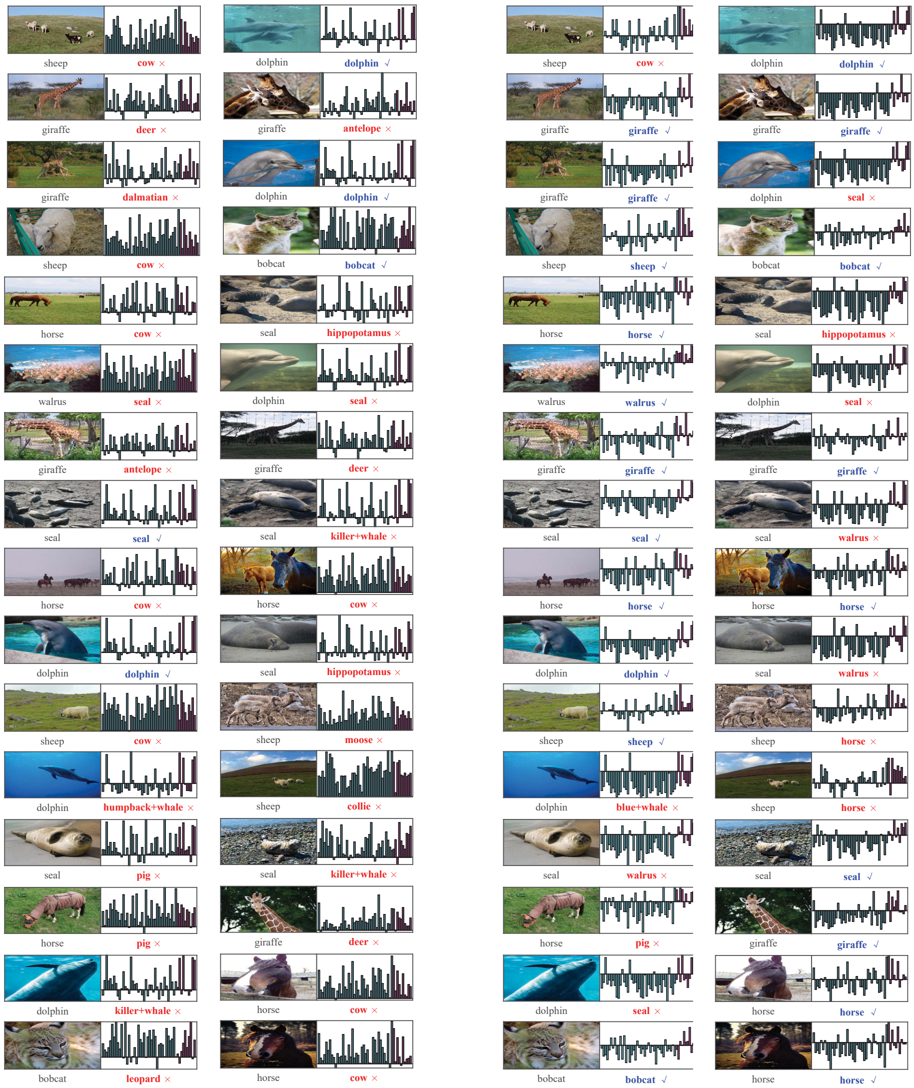
(b) GZSL: The 19 Randomly Selected Test Unseen Images for the 1*th* Time

Fig. 6: Visualizations of the attended parts for both seen/unseen test images on CUB, under GZSL. Different from the attended parts under ZSL, the discovered divergent and discriminative parts are #2-3, 5-6, 10. As can be seen from (a) and (b), RGEN can still 1) discover more divergent parts w.r.t. objects; 2) suppress background and redundant foreground regions (maximum mask values in parts #1, 4, 7-9 are all small and no similar masks exist among foreground parts #2-3, 5-6, 10); and 3) automatically align the order relationships of different parts (parts #2-3, 5-6, 10 are consistent w.r.t. different unseen/seen test class images). Note that, maximum mask values of part #3 and #6 are not very confident, however, the attended regions on these two columns of parts are still focused on the edge of the object. As such, we claim that these two columns of parts could be still beneficial to the final performance. We randomly select two times from both the unseen/seen test image set: (a) The 19 Randomly Selected Test Seen Images for the $1th$ time. (b) The 19 Randomly Selected Test Unseen Images for the $1th$ time. Zoom in four times to see details.



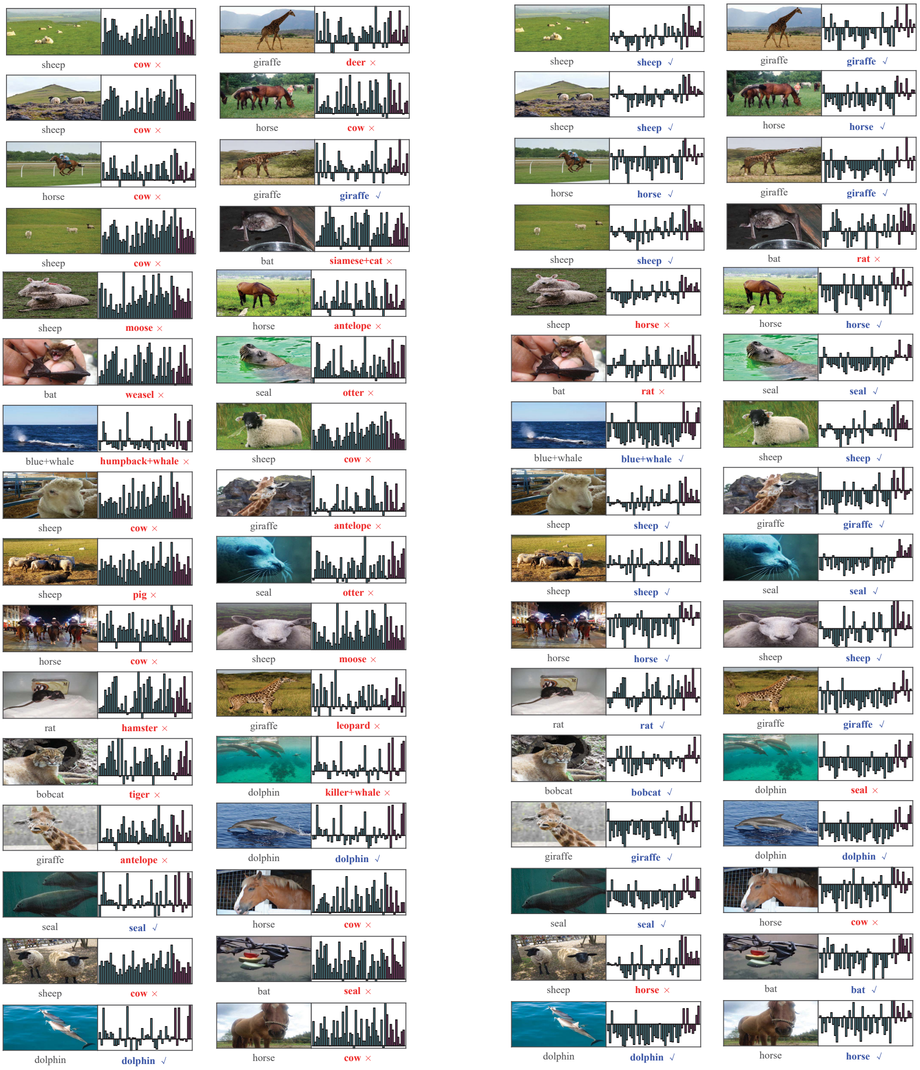
(a) GZSL: The 19 Randomly Selected Test Seen Images for the 2*th* Time (b) GZSL: The 19 Randomly Selected Test Unseen Images for the 2*th* Time

Fig. 7: Visualizations of the attended parts for both seen/unseen test images on CUB, under GZSL. Different from the attended parts under ZSL, the discovered divergent and discriminative parts are #2-3, 5-6, 10. As can be seen from (a) and (b), RGEN can still 1) discover more divergent parts w.r.t. objects; 2) suppress background and redundant foreground regions (maximum mask values in parts #1, 4, 7-9 are all small and no similar masks exist among foreground parts #2-3, 5-6, 10); and 3) automatically align the order relationships of different parts (parts #2-3, 5-6, 10 are consistent w.r.t. different unseen/seen test class images). Note that, maximum mask values of part #3 and #6 are not very confident, however, the attended regions on these two columns of parts are still focused on the edge of the object. As such, we claim that these two columns of parts could be still beneficial to the final performance. We randomly select two times from both the unseen/seen test image set: (a) The 19 Randomly Selected Test Seen Images for the 2*th* time. (b) The 19 Randomly Selected Test Unseen Images for the 2*th* time. Zoom in four times to see details.



(a) Baseline (b) RGEN

Fig. 8: Cyan and magenta bars are the predicted scores (before the softmax-layer in Baseline and RGEN models) on seen/unseen classes, respectively. Domain bias in (a) Baseline has been well addressed by our (b) RGEN, which further show the effectiveness of our RGEN under GZSL. Zoom in four times to see details. "√" indicates the input image is correctly classified as the ground-truth category. "×" indicates the input image is misclassified as the one beside this symbol.



(a) Baseline

(b) RGEN

Fig. 9: Cyan and magenta bars are the predicted scores (before the softmax-layer in Baseline and RGEN models) on seen/unseen classes, respectively. Domain bias in (a) Baseline has been well addressed by our (b) RGEN, which further show the effectiveness of our RGEN under GZSL. Zoom in four times to see details. "√" indicates the input image is correctly classified as the ground-truth category. "×" indicates the input image is misclassified as the one beside this symbol.