LOCL: Learning Object-Attribute Composition using Localization

Satish Kumar
satishkumar@ucsb.edu
ASM Iftekhar
iftekhar@ucsb.edu
Ekta Prashnani
eprashnani@ucsb.edu
B.S. Manjunath
manj@ucsb.edu

ECE Department,
University of California
Santa Barbara

1 Introduction

We propose a novel approach to learn unseen novel Object-Attribute (OA) associations in images. Our proposed approach has two steps – a robust weakly-supervised Localized Feature Extractor ($LFE(.)$), followed by a Composition Classifier ($CC(.)$). In this supplementary material, we provide the details of all the experiments done on LOCL to validate our design choices. We also provide details on the used datasets along with quantitative and qualitative results.

2 Implementation Details

Both the localized feature extractor $LFE(.)$ and Composition Classifier $CC(.)$ are trained on all the datasets. To train $LFE(.)$, an efficient contrastive pre-training framework is used with a margin distance of 1. As backbone image encoder, we use ResNet-50 [2] pre-trained on [14]. For text encoding we utilize text encoder similar to [14]. The Anchor Generator generates 576 valid anchor boxes. Corresponding to each of these anchor boxes, features $[\mathbf{f}_1^{anc}, \mathbf{f}_2^{anc}, ..., \mathbf{f}_{576}^{anc}]$ are pooled from $\mathbf{F}$. To generate ϕ according to Eq. 1, each $\mathbf{f}_j^{anc}$ is matched with semantic word embedding vector, and top 20 scores are labeled as 1 and rest as 0 as shown in Eq. 2 to create the pseudo label y. This is an empirically selected value, it covers almost all the object regions in the image. The **Region Proposal Network** generates proposal boxes and an objectness score corresponding to each proposal box. The number of proposal boxes are equal to the number of anchor boxes. Corresponding to these proposals boxes, features $[\mathbf{f}_1^p, \mathbf{f}_2^p, ..., \mathbf{f}_{576}^p]$ are pooled from feature map $\mathbf{F}$.

Contrastive loss is used for pre-training. The cosine distance d_k is computed between each anchor feature $\mathbf{f}_k^{anc}$ and $\mathbf{f}_k^p$, the total number of features are 576 for anchors and 576 proposals. The pseudo label y is of length 576. y_k is equal to 1 where the $\mathbf{f}_k^{anc}$ feature have

potential object. The scaling parameters for contrastive loss α and classification loss β are set to $0.6, 0.4$ respectively. The network is trained for 100 epochs, convergence is observed around 45 epochs, based on that, early stopping is done at 50 epochs. The learning rate starts with $1e^{-5}$ with decay of 0.1 after every 10 epochs. The batch size is set at 24. The optimizer used is *Adam* optimizer. The region proposal branch of the network learns to select features from regions where the objects are present. During training, we restrict the learning rate of linear projection layer of f_{ao}^{txt} to a low value to stabilize the region proposal branch.

Compositional Classifier $CC(.)$: The ability to learn individual representation of O-A in visual domain is crucial for transferring knowledge from seen to unseen O-A associations. Existing SOTA works [8, 9, 11, 15, 17] use homogeneous features from whole image as without localizing the object, they ignore the discriminative visual features of object and its attributes. Our Composition Classifier network $CC(.)$ leverages the distinctive features extracted by $LFE(.)$ to predict the object and its corresponding attribute. It is challenging to associate right attribute with the object by using homogeneous features, as there can be interference from prominent confounding elements. $CC(.)$ takes as input, the top 10 pooled features $[\mathbf{f}_1^p, \mathbf{f}_2^p, ..., \mathbf{f}_{10}^p]$ from pre-trained $LFE(.)$ sorted in descending order based on objectness score $\hat{o} = [\hat{o}_1, \hat{o}_2, .., \hat{o}_{10}]$. Each block in $CC(.)$ consists of two fully connected layer with ReLU activation. The initial learning rate for $CC(.)$ network is set to $1e^{-3}$ with a decay of 0.1 after every 7 epochs. We observed that fine tuning $LFE(.)$ with a lower learning rate of $1e^{-6}$ while training $CC(.)$ performed better than freezing it. The batch size used is 32. All the experiments are done on a single nvidia V100 Tesla.

3 Results & Analysis

We report Top-1 accuracy in seen and unseen classes and accuracy in detecting objects and attributes. LOCL achieve best accuracy in individual detection of objects and attributes as shown in Table 1. This is interesting as our simple $CC(.)$ do not a have dedicated object detector similar to SymNet [7]. In MIT-States [5] LOCL outperforms SOTA method by 12% on object detection accuracy and 19% attribute detection. In UT-Zappos [19] LOCL's performance is slightly better than SOTA methods since each image has one dominant object with clear white background. Moreover the performance improvement is significant when it comes to more challenging and realistic dataset CGQA [11].

Methods	MIT-States [5]		UT-Zappos [19]		CGQA [11]	
	Object	Attribute	Object	Attribute	Object	Attribute
Attop [12]	21.1	23.6	38.9	69.6	8.3	12.5
LabelEmbed [11]	23.5	26.3	41.2	69.2	7.4	15.6
TMN [13]	23.3	26.5	40.8	69.9	9.7	20.5
SymNet [7]	26.3	28.3	40.5	71.2	14.5	20.2
CompCos [9]	27.9	31.8	44.7	73.5	-	-
CGE [11]	30.1	34.7	48.7	76.2	15.2	30.4
LOCL (Ours)	**42.7**	**53.4**	**49.4**	**79.3**	**28.7**	**35.1**

Table 1: Performance comparisons on detecting individual objects and attributes. LOCL outperforms all compared methods with a significant margin.

4 Datasets

The splits used on all the datasets are as follows. MIT-States [5] has a total of 53,000 images with 245 objects and 115 attributes. The splits for MIT-States dataset have 1262 object-attribute pairs (34,000 images) for the training set, 600 object-attribute pairs (10,000 images) as the validation set and 800 pairs (12,000 images) as test set. All the images in MIT-states dataset are of natural objects collected using an older search engine with limited human annotation causing a significant label noise [1]. UT-Zappos [19] has 29,000 images of shoes catalogue. The splits used are of 83 object-attribute pairs (23,000 images) for the training set, 30 object-attribute pairs (3,000 images) for the validation set and 36 pairs (3,000 images) for test set. The images in UT-Zappos [19] dataset are not really entirely a compositional dataset as the attributes like *Faux Leather vs Leather* are material differences but not specifically any visual difference [8]. Also, the simplicity of the images (one object with white background) makes it unsuitable to work in natural surroundings where the object of interest has interference confounding elements in the scene. These splits are selected following previous works [8, 13]. The third dataset used is Compositional-GQA (CGQA) dataset [8, 11]. It has 453 attributes and 870 objects. The splits for CGQA have 5592 object-attribute pairs (26,000 images) for training set, 2292 pairs (7,000 images) for validation set and 1811 pairs (5,000 images) for testing set. These splits are as proposed by [8]. The CGQA dataset have images curated from visual genome dataset [6] which comprises of images from natural and realistic settings. Most of the images in CGQA have an object of interest with confounding elements in the background, that makes it an extremely challenging dataset to evaluate CGQA models.

5 Ablation Study

In this section, we discuss about the additional design choices for training of the Localized Feature Extractor $LFE(.)$ and composition classifier $CC(.)$.

5.1 Number of Proposals:

Table 2 shows the selection criterion of number of proposal selected from pre-trained $LFE(.)$ the goes as input to $CC(.)$. With $r < 10$, the proposals features miss regions of the object, which leads to poor performance. While when $r > 10$, more background features are picked that suppress the prominent object and lead to drop in prediction quality.

# of proposals	MIT-States		
	Seen	Unseen	AUC
5	32.1	33.6	7.2
10	**35.3**	**36.0**	**7.7**
15	35.3	35.9	6.9
20	27.6	28.4	6.5

Table 2: Performance of LOCL as we select different number of top r proposals from pre-trained LFE. Best performance is observed with *r=10*. With $r > 10$, more background features are picked that suppress the prominent objects.

5.2 Object\Attribute Refinement:

Table 3 shows refinement operations done on visual features $\mathbf{f}_p^{all}$ as shown in Eq.6 in the main paper. The multiplication operations generates more selective information and suppresses the redundant information as compared to concatenation and addition operation [4, 16].

Method	MIT-States [5]		
	Seen	Unseen	AUC
Addition	28.5	29.6	6.6
Multiplication	**35.3**	**36.0**	**7.7**
Concatenation	32.7	33.1	7.2

Table 3: Performance of compositional classifier with different refinement operations.

5.3 Pre-training $LEF(.)$ with object embeddings:

As discussed in section 3.1 of the main paper, we use OA pair name $< Blue, Bird >$ as input during the pre-training of $LEF(.)$. We also test with using only the object names $< Bird >$ as the input. We observe 3% drop in accuracy as compared to OA pair names in the unseen category as shown in Table 4. This is expected as the text embeddings generated by the text encoder are more meaningful and have closer representation with the visual features when we provide a complete description of the object in the image i.e. OA pair name.

Names used	MIT-States [5]				
	Seen	Unseen	Obj	Attr	*AUC*
Obj-Attr	**35.3**	**36.0**	**42.7**	**53.4**	**7.7**
Obj	32.5	32.8	37.4	41.9	7.1

Table 4: Performance of the network in MIT-States [5] with different names used as input to the text encoder while pre-training $LFE(.)$. **Bold** numbers are the best performance settings. The network performs well with $Obj-Attr$ names as input compared to just Obj names.

5.4 Number of Pseudo Labels:

For creating pseudo labels y during pre-training, as mentioned in section 3.1 equation 4, we assign value 1 to top 20 indexes and rest are assigned 0. The equation is:

$$\phi = [\phi_1, \phi_2, .., \phi_k, .., \phi_n] \tag{1}$$

$$y = \begin{cases} 1 & argsort(\phi)[0:l] \\ 0 & for\ all\ other\ indexes \end{cases} \tag{2}$$

where $y = [y_1, y_2, .., y_k, .., y_n]$, 20 anchors are selected based on cosine similarity score ϕ (equation 2 in main paper). They are assigned with label 1 in y and rest are assigned 0 as shown above with Eq. 2. Here each y_k represents the presence/absence of object of interest regions in the input image. We experiment with different values for number of potential objects. As shown in Table 5, the overall performance of the model drops if we pick a number greater than or less than 20. This is because for smaller value, the $LFE(.)$ is penalized for detecting even the right regions of interests and for larger value than 20, we are learning

information from confounding elements from the background where the object may/may not be present.

#Pseudo Labels	MIT-States [5]				
	Seen	Unseen	AUC	Obj	Attr
10	31.5	27.9	5.2	27.9	31.5
15	33.8	34.1	6.5	28.4	30.8
20	**35.3**	**36.0**	**7.7**	**42.7**	**53.4**
25	29.1	29.6	6.1	31.5	34.6

Table 5: Performance of the network in MIT-States [5] with different number of region of interest while pre-training $LFE(.)$. **Bold** numbers are the best performance settings. Here # is "Number of".

5.5 Margin distance for contrastive loss:

For pre-training $LEF(.)$ with contrastive loss, we use a margin distance of 1 as shown in equation 5 in the main paper. We experimented with different distances for the margin for MIT-states [5] dataset. We achieved best performance at a margin of 1. The experimental evaluation with different margin distance is shown in Table 6. Our observations of is that with bigger margin, the network start clustering features from those regions also, which have object of interest along with a significant section of background regions. This leads to drop in attribute detection accuracy.

Margin	MIT-States [5]				
	Seen	Unseen	AUC	Object	Attribute
0.5	29.6	30.4	5.2	30.2	47.3
1.0	**35.3**	**36.0**	**7.7**	**42.7**	**53.4**
3	34.1	33.9	6.5	41.1	46.8
7	25.3	26.5	4.8	37.3	38.9

Table 6: Performance of pre-training the $LFE(.)$ using different margin distance for contrastive learning. We achieve best performance when margin is 1. For higher margin, $LFE(.)$ cluster features of object of interest which have significant region of background/confounding regions also. Leaning to poor performance.

5.6 Scaling parameters of loss function:

While pre-training, we combine contrastive loss and binary cross entropy loss using scaling parameters α and β. The equation is:

$$\mathcal{L}_{total} = \alpha * \mathcal{L}_{CON} + \beta * \mathcal{L}_{BCE}(o, \phi), \tag{3}$$

where, $\mathcal{L}_{CON}$ is the contrastive loss and $\mathcal{L}_{BCE}$ is the binary cross entropy loss. We test for different values α and β as shown in Table 7. It appears giving a bit more weight to the contrastive loss helps $LFE(.)$ to extract better localized features.

6 Qualitative Results

We add more qualitative results for unseen novel composition with top-1 prediction in Figure 1. The examples are presented from datasets : CGQA [11], MIT-States [5], and UT-

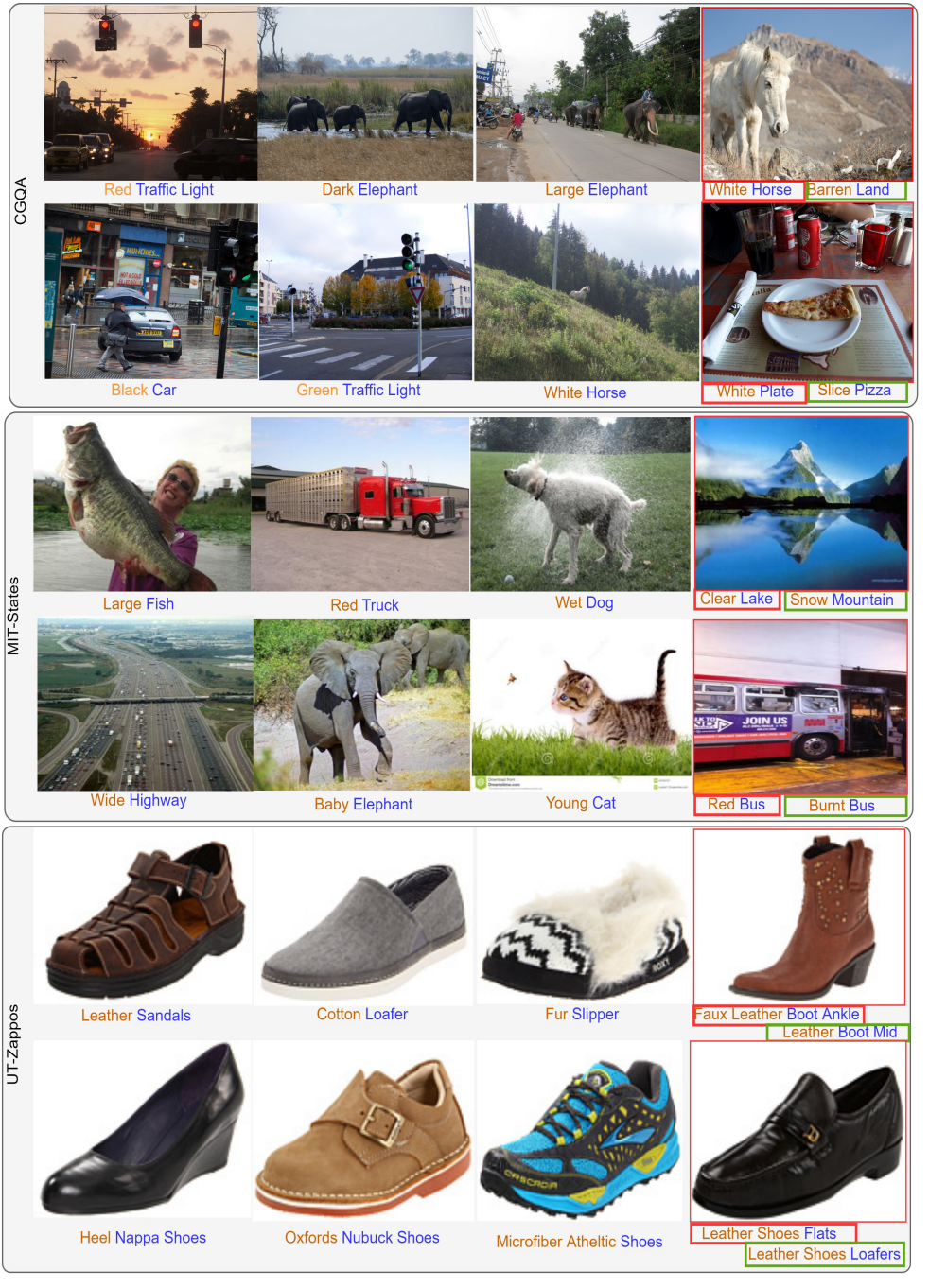

Figure 1: Qualitative results of LOCL. Left three columns show correct predictions from our network. Rightmost column shows missed predictions, here, ground truth labels are marked with green box and our predictions are marked in red box. The datasets contain only one OA pair and our predictions though visually correct, do not match with the ground-truth OA in these cases.

Parameters		MIT-States [10]				
α	β	*Seen*	*Unseen*	*AUC*	Obj	Attr
0.3	*0.7*	30.6	32.5	6.9	30.9	33.1
0.4	*0.6*	35.1	35.4	7.5	36.4	39.9
0.5	*0.5*	34.9	35.8	7.7	40.8	49.3
0.6	***0.4***	**35.3**	**36.0**	**7.7**	**42.7**	**53.4**
0.7	*0.3*	32.7	33.7	7.1	33.0	35.2

Table 7: Performance of the network with different scaling parameters of the loss function during pre-training. **Bold** numbers are the best performance settings.

Zapos [19]. The order of the datasets is in decreasing order of the clutter in the images. As can be seen that in the CGQA dataset, the images contains object of interest with lot of confounding elements creating background clutter. MIT-States [5] is also of natural images. However, most of the images have a dominant object. On the other hand, in UT-Zappos [19] all the images contain a single object with clear white background. This shows the complexity and the challenges of CGQA dataset compared to the existing ones.

The first three columns represent the examples where our model is making the right predictions. The last column in each dataset shows examples where our model makes the visually correct prediction. However, it does not match with the ground truth label of object and attribute. Our model is selecting object of interest, and it is creating the right attribute-object associations. For example in case of fourth row on the rightmost column, our prediction of the object is right but the image contains multiple attributes, while the ground truth contains only one OA pair. This put an artificial limitation on the evaluation metric even when the predictions are perceiveably correct.

BMP-Net [18] achieves state-of-the-art (SOTA) performance in seen classes of MIT-States [10] and UT-Zappos [19]. However, their sub-optimal performance in unseen classes indicates a bias towards seen classes. To further investigate this bias, we evaluate BMPNet on the challenging CGQA dataset [11]. We utilize the official repository provided by the authors for this evaluation and report performance in the same matrices used for other datasets. In Table 8, we can observe LOCL outperforms BMPNet in all category. Especially in unseen classes, LOCL achieves more than double accuracy than BMPNet. This poor performance indicates a seen class bias of BMPNet. This bias is mainly due to creating the graph network with a large number of seen and plausible OA pairs. More discussion on this phenomenon is available in section 4.4 in the main paper. Moreover, LOCL is very efficient and utilizes only $\sim$ 5GB memory for training in the large-scale dataset CGQA. Current graph-based SOTA networks CGE [11] ($\sim$ 10GB), BMPNet [18] ($\sim$ 40GB) utilize much higher GPU memory for the same batch size in CGQA dataset. Therefore, LOCL is suitable for training on large scale challenging CZSL datasets.

Methods	CGQA [11]		
	Seen	Unseen	AUC
BMP-Net [18]	29.1	11.7	2.7
LOCL (Ours)	**29.6**	**26.4**	**4.2**

Table 8: Performance comparison on CGQA [11] dataset. LOCL significantly outperform BMP-Net [18] in a challenging (significant background clutter) dataset. The performance of LOCL shows the effectiveness of **LEF** in unseen OA associations.

References

[1] Yuval Atzmon, Felix Kreuk, Uri Shalit, and Gal Chechik. A causal view of compositional zero-shot recognition. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1462–1473. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/1010cedf85f6a7e24b087e63235dc12e-Paper.pdf.

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

[3] Drew A Hudson and Christopher D Manning. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709, 2019.

[4] ASM Iftekhar, Satish Kumar, R Austin McEver, Suya You, and BS Manjunath. Gtnet: Guided transformer network for detecting human-object interactions. *arXiv preprint arXiv:2108.00596*, 2021.

[5] Phillip Isola, Joseph J Lim, and Edward H Adelson. Discovering states and transformations in image collections. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1383–1391, 2015.

[6] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73, 2017.

[7] Yong-Lu Li, Yue Xu, Xiaohan Mao, and Cewu Lu. Symmetry and group in attribute-object compositions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11316–11325, 2020.

[8] Massimiliano Mancini, Muhammad Ferjad Naeem, Yongqin Xian, and Zeynep Akata. Learning graph embeddings for open world compositional zero-shot learning. *arXiv preprint arXiv:2105.01017*, 2021.

[9] Massimiliano Mancini, Muhammad Ferjad Naeem, Yongqin Xian, and Zeynep Akata. Open world compositional zero-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5222–5230, 2021.

[10] Ishan Misra, Abhinav Gupta, and Martial Hebert. From red wine to red tomato: Composition with context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1792–1801, 2017.

[11] Muhammad Ferjad Naeem, Yongqin Xian, Federico Tombari, and Zeynep Akata. Learning graph embeddings for compositional zero-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 953–962, 2021.

[12] Tushar Nagarajan and Kristen Grauman. Attributes as operators: factorizing unseen attribute-object compositions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 169–185, 2018.

[13] Senthil Purushwalkam, Maximilian Nickel, Abhinav Gupta, and Marc'Aurelio Ranzato. Task-driven modular networks for zero-shot compositional learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3593–3602, 2019.

[14] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021.

[15] Frank Ruis, Gertjan Burghours, and Doina Bucur. Independent prototype propagation for zero-shot compositionality. *arXiv preprint arXiv:2106.00305*, 2021.

[16] Oytun Ulutan, ASM Iftekhar, and Bangalore S Manjunath. Vsgnet: Spatial attention network for detecting human object interactions using graph convolutions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13617–13626, 2020.

[17] Guangyue Xu, Parisa Kordjamshidi, and Joyce Y Chai. Zero-shot compositional concept learning. *arXiv preprint arXiv:2107.05176*, 2021.

[18] Ziwei Xu, Guangzhi Wang, Yongkang Wong, and Mohan S Kankanhalli. Relation-aware compositional zero-shot learning for attribute-object pair recognition. *IEEE Transactions on Multimedia*, 2021.

[19] Aron Yu and Kristen Grauman. Semantic jitter: Dense supervision for visual comparisons via synthetic images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5570–5579, 2017.