Cleaner categories improve object detection and visual-textual grounding

Davide Rigoni[1,2](✉), Desmond Elliott[3], and Stella Frank[3,4]

[1] Department of Mathematics "Tullio Levi-Civita", University of Padua
[2] Bruno Kessler Foundation
[3] Department of Computer Science, University of Copenhagen
[4] Pioneer Center for AI, Denmark
davide.rigoni.2@phd.unipd.it {de,stfr}@di.ku.dk

Abstract. Object detectors are core components of multimodal models, enabling them to locate the region of interest in images which are then used to solve many multimodal tasks. Among the many extant object detectors, the Bottom-Up Faster R-CNN [39] (BUA) object detector is the most commonly used by the multimodal language-and-vision community, usually as a black-box visual feature generator for solving downstream multimodal tasks. It is trained on the Visual Genome Dataset [25] to detect 1600 different objects. However, those object categories are defined using automatically processed image region descriptions from the Visual Genome dataset. The automatic process introduces some unexpected near-duplicate categories (e.g. 'watch' and 'wristwatch', 'tree' and 'trees', and 'motorcycle' and 'motorbike') that may result in a sub-optimal representational space and likely impair the ability of the model to classify objects correctly. In this paper, we manually merge near-duplicate labels to create a cleaner label set, which is used to retrain the object detector. We investigate the effect of using the cleaner label set in terms of: (i) performance on the original object detection task, (ii) the properties of the embedding space learned by the detector, and (iii) the utility of the features in a visual grounding task on the Flickr30K Entities dataset. We find that the BUA model trained with the cleaner categories learns a better-clustered embedding space than the model trained with the noisy categories. The new embedding space improves the object detection task and also presents better bounding boxes features representations which help to solve the visual grounding task.

Keywords: Object Detection, Visual Genome, Bottom-Up, Data Cleaning, Label Cleaning, Object Ontology

1 Introduction

Object detection is the task of locating and classifying the objects depicted in an image [32]. This is a core task in the field that is used whenever there is the need to localize and recognize objects in images, such as when an autonomous driving car needs to recognize road signs, people, and objects in the streets.

Beyond computer vision, object detectors are the cornerstone of multimodal vision and language (V&L) tasks, which require jointly reasoning over visual and linguistic input. Indeed, in order to reason about the objects in the image, it is first necessary to identify them. Examples of such tasks are the referring expression recognition and visual grounding [42, 7, 63, 17, 40, 22], visual question answering [2, 43, 68], visual-textual-knowledge entity linking [13, 11, 12] and image-text retrieval [34, 24, 66, 29, 55]. In these V&L tasks, the object detector is used as a static black-box feature extractor. Therefore, it needs to be accurate and comprehensive in order to support the downstream multimodal tasks.

The Bottom-Up Faster R-CNN [1] (BUA) object detector is one of the most commonly-used black box object detectors in the field. Within the V&L literature, it is the defacto standard feature extractor used to represent the visual input [16]. BUA is pretrained on the Visual Genome dataset [25] to detect 1600 objects, e.g. "chair", "horse", "woman", and also to predict their attributes, e.g. "wooden", "brown", "tall". Both the category and attribute set are derived from the freely annotated region descriptions in the Visual Genome dataset, rather than using pre-defined categories like in ImageNet [10] or COCO [31]. Anderson et al. did attempt to filter the categories and attributes to prevent near-duplicates, however, the resulting 1600 categories are still imperfect. There are synonymous categories ("wrist watch", "wristwatch"), categories representing single and plurals of the same concepts ("apple", "apples"), ambiguous, difficult to differentiate, categories ("trousers", "slacks", "chinos", "lift"), and categories that actually represent attributes such as "yellow" or "black". We argue that having to predict these noisy categories is likely to prevent the object detector from supporting downstream tasks well.

In this work, we propose a new set of categories that can be used to train the BUA object detector on the Visual Genome dataset. The new set is the result of a cleaning process performed manually by a native English speaker. Starting from the original 1600 noisy categories, the ambiguous categories were merged to build the final set of 878 clean categories. We then use these clean categories to re-train the BUA object detector. In addition to evaluating its object detection performance, we analyze the model's feature embedding space, and evaluate the benefits of using its features in a downstream referring expression comprehension grounding task. In our experiments, the BUA model trained with the cleaned categories detects objects better, and, examining its feature space representation, we find out that it learns a better-clustered embedding space than the model trained with the original noisy categories. The new embedding space produces better bounding boxes feature representations, which in turn can improve performance on a downstream visual-textual grounding task.

The contributions of this paper are summarized as follows:

1. starting from the 1600 noisy categories developed by [1], we propose a cleaner set of 878 categories with less noise and fewer near-duplicates;
2. we show that a BUA detector trained on these cleaned categories improves object detection performance and produces a better visual embedding space compared to using the original noisy categories;

3. finally, we show that using the new detector as a black-box feature extractor can improve performance on a downstream visual-textual grounding task.

2 Related Work

This paper relates to (a) work that adopts the Bottom-Up model [1] for the detection of objects depicted in images, especially for multimodal downstream tasks, and (b) work that addresses learning neural networks with noisy labels. We describe the Bottom-Up model itself in more detail in Section 3.

2.1 Bottom-Up for Object Detection

Many object detectors exist [39, 67, 9, 49, 56, 59, 57, 64, 58, 50], that differ according to their ability to detect objects in the image, the computing power required for their use, and their ability to recognize a large set of different objects[1, 38]. An object detector should be able to identify many different objects [62] and classify them correctly. The appeal of BUA features lies in part in the large number of object categories. Nevertheless, the increase in the number of objects to be recognized leads to a more challenging classification problem.

Starting with [1], in which the extracted object detector bounding boxes were used as input to a Visual Question Answering (VQA) model, much work on VQA adopted the BUA model as object detector [6, 61, 26, 5, 45, 69, 20, 54]. BUA features have also been used for the Referring Expression Comprehension task [62, 41, 23, 53, 52, 23]. In addition, many recent large pretrained Vison and Language models use BUA features as their visual representations [27, 47, 30, 48, 33, 15, 4]. These models are used as the starting point for a wide variety of multimodal tasks, including image description, VQA, natural language visual reasoning, referring expression comprehension, etc [35, 21, 16].

All these works directly depend on the quality of the objects detected by the BUA model. Incorrect identification and/or classification of objects may have major repercussions in the resolution of downstream tasks, making it important to analyze in more detail the labels used to train the BUA model.

2.2 Noisy Label Sets

This work, aiming to improve data quality by improving label quality, is related to the branch of research area addressing noisy label effects during neural network training. However, most of this work addresses the problem of badly labeled data, i.e. noise at the instance level (see [46] for a recent survey).

We are interested in the problem of bad or noisy labels, rather than noisy data. [36] show that their framework for estimating noise in data labelling can also identify 'ontological issues' with the labels themselves. Removing duplicate labels during training improves performance on ImageNet classification, in line with the object detection improvement we find in this paper. [3] identify and correct label issues in ImageNet for better, more robust model evaluation and comparison; removing 'arbitrary' label distinctions ensures models are not rewarded for overfitting to spurious noise. [51] aim to discover a 'basic level' label set, i.e. the labels corresponding to the human default or basic level categories, by merging labels that are often confused. They find that training an image classifier on these categories can improve downstream image captioning and VQA.

3 Recap: Bottom-Up Faster R-CNN

The Bottom-Up [1] model is based on the Faster R-CNN [39] object detector devised to recognize instances of objects belonging to a fixed set of pre-defined categories and localize them with bounding boxes. Faster R-CNN initially uses a vision backbone, such as ResNet [18] or a VGGNet [44], to extract image features from the image. Then Faster R-CNN applies a Region Proposal Network (RPN) over the input image, that predicts a set of class-agnostic bounding box proposals for each position in the image. The RPN aims to detect all the bounding boxes that contain an object, regardless of what the object is. Then, for each detected bounding box proposal, Faster R-CNN predicts a class-aware probability score and a refinement of the bounding box coordinates to better delimit the classified object. The Faster R-CNN multi-task loss function contains four components, defined over the classification and bounding box regression outputs for the Region Proposal Network and the final bounding boxes refinement.

The BUA object detector initializes its Faster R-CNN backbone weights from a ResNet-101 [19] model pre-trained on the ImageNet [10] dataset for solving the image classification task. The model is trained on the Visual Genome [25] dataset to predict 1600 different objects. Since the Visual Genome dataset also annotates a set of attributes for each bounding box in addition to the category it belongs to, the BUA model adds an additional trainable module for predicting attributes (in addition to object categories) associated with each object localized in the image. For this reason, the BUA model adds a multi-class loss component to the original Faster R-CNN losses to train the attribute predictor module.

The 1600 categories used to train the BUA model were set by [1]. The Visual Genome dataset annotations consist of image regions associated with region descriptions (natural language strings) and the attributes of the object depicted in it. [1] extract category labels from the region descriptions, but their procedure is underspecified (for example, it is unclear if they used a part-of-speech tagger to extract nouns and adjectives as labels for objects and attributes). They filtered the original set of 2500 object strings and 1000 attribute strings based on object detection performance, resulting in a set of 1600 categories and a set of 400 attributes. However, the remaining set of categories is still noisy. It contains plurals and singular of the same concepts, such as "dog" and "dogs", overlapping categories such as "animal", "cat", and "dog". Moreover, it contains near-duplicate categories such as "motorcycle" and "motorbike", unhelpful distinctions like "lady" and "woman", labels representing attributes such as "yellow" and abstract notions like "front". These noisy labels may result in a sub-optimal representational space and likely impair the ability of the model to classify objects correctly. Given that several labels equivalently express the same meaning, whenever the model needs to predict a category for an object appearing in the image, the model needs to split its predicted probabilities among all equivalent categories. This probability split occurs not only when two or more categories express the same meaning (e.g. “hamburger” and “burger”) but also when the meanings expressed by the categories overlap substantially, such as the categories “pants”, “trousers”, and “slacks”.

4 Cleaning the Visual Genome Category Set

In this paper, we propose a new set of categories to use for training the BUA object detector. This new label set is the outcome of a cleaning process applied to the 1600 original categories by the authors of this paper, which include native English speakers. This process aimed to combine ambiguous and low-frequency categories together. During the cleaning process, the categories were joined together according to the following principles:

1. **Plurals**: singular and plurals categories, such as “giraffe” and “giraffes”. In most instances, these annotations represent the same concept and should be treated as the singular category. This led to 258 category merges.
2. **Tokenization**: categories with and without spaces, such as “wrist watch” and “wristwatch”, should be treated as the same category. This resulted in 29 category merges.
3. **Synonyms**, such as “microwave” and “microwave oven”, “hamburger” and “burger”, express similar concepts with minor differences that are usually not important. Often, as in “microwave oven”, these are compound phrases that can be identified automatically, though it is important to verify them manually (e.g. “surf” and “surf board” should not be merged).
4. **Over-specific** categories with substantial annotator disagreement where several words are used interchangeably, e.g. “pants”, “trouser”, “sweatpants”, “jean”, “jeans”, and “slacks”.

However, during the cleaning process, it was not always clear when to merge the categories since: (i) some categories are inherently ambiguous, such as “home”; (ii) some categories are abstract and don’t have the meaning of a concrete object, such as “items”, “front”, “distance”, “day”; (iii) some categories represent attributes rather than objects, such as “yellow” and “black”.

For some ambiguous labels like ‘lot’ or ‘lift’, visual inspection of the labelled images showed that within VG, these labels were used mostly to refer to one concept: “lot” usually showed car parking and was merged with “parking lot”, similarly “lift” was merged with “ski lift”. In other cases, no single meaning predominated and these labels were left un-merged (e.g. ‘stand’ was not merged with either ‘baseball stand’ nor ‘tv stand’). The abstract and attribute categories were also left as they were. In this way, the adopted cleaning process defines a surjective function that maps the original labels set to cleaner labels set.

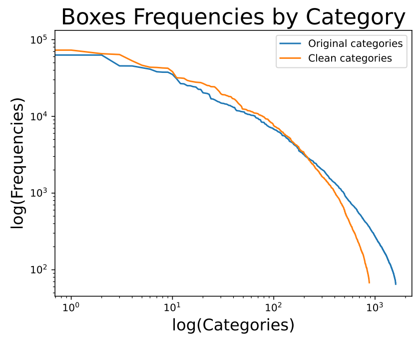

Fig. 1: LogLog plots of objects frequencies for each category. The frequencies are calculated on the training set annotations. The distribution of the original categories is in blue, and the new categories are in orange. The cleaning process did not generate high-frequency categories and at the same time removed many low-frequency categories.

The cleaning process produces a new set of 878 categories from the original 1600 categories (Appendix 1). Figure 1 shows frequencies of objects appearing in the Visual Genome training split, where objects are either labeled according to the original label set (in blue) or the new cleaned label set (in orange). The new labels lead mostly to the removal of many low-frequency categories in the long tail, rather than creating new very frequent categories.

5 Experimental Setup

We train a BUA object detector matching the procedure of Anderson et al. [1], except that we use the new clean categories as object labels instead of the original noisy categories.

5.1 Datasets and Evaluation Metrics

Following [1], the training and test data for the models is the Visual Genome (VG) dataset [25]. It is a multipurpose dataset that contains annotations of images in the form of scene graphs that form fine-grained descriptions of the image contents. It supplies a set of bounding boxes appearing in the image, with labels such as objects and persons, together with their attributes, such as color and appearance, and the relations between them. The original VG labels were converted to object labels by [1], as described in Section 3. We note here that our BUA model is trained only using the VG training split, unlike some pre-trained models available, e.g. in the MILVLG repository, which use both training and validation splits for training.

To assess the object detectors' performance, we use the Mean Average Precision (AP) metric, which is the standard metric for measuring the accuracy of object detectors such as Faster R-CNN [39]. All evaluation results presented in this work are obtained on the VG test split. Average precision uses a Intersection over Union threshold of 0.5 to determine whether the predicted bounding box is sufficiently similar to the gold region. We distinguish between 'macro' and 'micro' (also known as 'weighted') AP: MacroAP weights each category uniformly (macro-averaging class-wise precision) while MicroAP weights each category by the number of items in the category (equivalent to micro-averaging over all items, regardless of class). MacroAP will emphasize the effect of small categories, while MicroAP will be dominated by the most frequent categories.

Precision is indirectly affected by the number of categories in the label set: e.g. a random baseline over 100 categories will perform worse than a baseline over 10 categories. Since our objective in this paper is to compare models with different numbers of categories, this is an unavoidable confound. To mitigate against it, for the original model, which predicts labels in the original label set, we map its predictions to the clean label set. For example, if the model predicts 'motorcycle' in the original label set, this prediction gets mapped to the same category ID as the model's 'motorbike' predictions, because these two labels have been collapsed in the clean label set. This results in mapped predictions with the same number of categories as the clean label set predictions, which means that comparison between label sets is fairer. However, this procedure also removes all errors due to confusing the two labels that have been merged in clean (e.g. if the original gold label for the 'motorcycle' prediction was 'motorbike', this incorrect prediction is now counted as correct), which makes it a very strict evaluation.

5.2 Random Baseline

We also compare against a BUA detector trained with a randomly merged category set. The randomly merged set was created by randomly selecting pair of categories in the original set to combine until we reached the same number of categories adopted in the clean set (i.e. 878). This procedure leads to a distribution of category sizes that is very similar to the clean label set, see Appendix 1. However, the randomly merged categories will include semantically very distinct objects, e.g. bananas and motorcycles are in the same category. This allows us to separate the effect of having cleaner categories from the effect of simply having fewer categories.

5.3 Implementation Details

For the development of this work, we used the code available in the MILVLG[5] repository, which is a Pytorch implementation of the original Caffe[6] model. In particular, the MILVLG code allows to train, evaluate, and extract bounding boxes from images using both the Detectron2 framework[7] as well as the original

[5] https://github.com/MILVLG/bottom-up-attention.pytorch
[6] https://github.com/peteanderson80/bottom-up-attention
[7] https://github.com/facebookresearch/detectron2

Table 1: BUA object detection results on the Visual Genome dataset. The model trained on the clean categories, "BUA Clean", achieves better object detection performance than the model trained on the original categories. "BUA Original→Clean-878" and "BUA Original→Random-878" are results from models trained on the original categories whose predictions are mapped to clean and random label set respectively, to match label set size (878 labels in both cases)

		Visual Genome (%)	
Model	**Implementation**	**MacroAP50↑**	**MicroAP50↑**
BUA Original	Caffe	9.37	15.14
BUA Original	PyTorch	9.10	15.93
BUA Original→Clean-878	PyTorch	10.72	17.34
BUA Clean	PyTorch	11.01	17.60
BUA Original→Random-878	PyTorch	9.49	15.79
BUA Random	PyTorch	9.46	15.61

Caffe model weights. When not explicitly indicated, we use BUA implemented with Detectron2. Between 10 and 100 bounding boxes are extracted for each image in input. We use the default MILVG hyper-parameters, apart from setting the batch size to 8, and training only on the training data split. We did not re-tune the model hyper-parameters when training on the new label set and used the same default hyper-parameters from the model trained on the original 1600 categories. The object detectors are trained for 180K iterations. All experiments were performed in a distributed parallel system using a V100 32GB GPU.[8]

6 Experiments

Our experiments compare BUA models trained on the new smaller label set with the original BUA model using the original label set. We compare these two models in terms of performance on the original object detection task, the properties of the embedding space learned by the detector, and the utility of the features in a visual grounding task on the Flickr30K Entities dataset. We expect the removal of label ambiguity in the new label set to lead to better performance on object detection and visual grounding.

6.1 Object Detection

We test object detection on the Visual Genome test set: see Table 1. The model trained on the new labels, BUA Clean, outperforms the BUA Original model by nearly two points on macro and micro AP.

To check how much of this improvement is due to simply having a smaller label set, we also compare both against the random (i.e. BUA Random) baseline

[8] https://github.com/drigoni/bottom-up-attention.pytorch

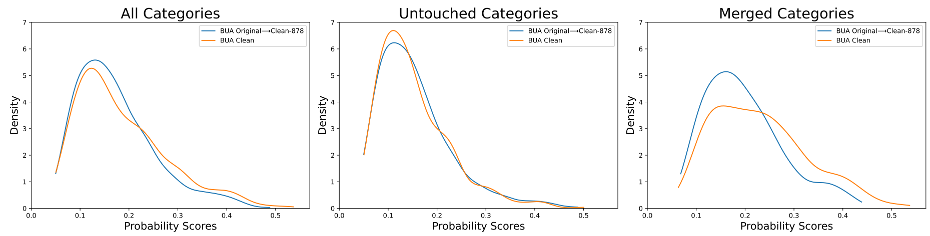

Fig. 2: KDE plots for the probability values of the argmax category predicted by the model. The plots on the left consider all the categories, the plots in the center consider just the categories that we did not merge during the cleanup process (i.e. "Untouched"), and the last plots on the right consider only the merged categories. Overall, the cleaned categories lead to higher confidence values than the original categories.

(where categories were iteratively merged to the same number of labels as the clean set) and against the same original predictions, but with predicted labels mapped to the clean set (e.g., predictions for 'egg' and 'eggs' are mapped to the same label, as in the clean set). The BUA Random results are slightly worse than the BUA Original model, indicating that fewer labels on their own are not enough to micro or macro AP. Mapping the original predictions to the new labels improves both metrics, indicating that many of the mistakes in the BUA Original model are due to confusion between labels that are merged in the clean set. However, performance does not reach the level of BUA Clean model, demonstrating that using better labels at training time is important. Since we see this improvement in both micro and macro AP, the new labels do not only improve frequent categories (reflected in MicroAP) or infrequent categories (MacroAP).

Figure 2 shows how noise in the category set affects the prediction confidence of the model. By 'prediction confidence', we mean the probability assigned to the argmax category predicted by the model when it detects an object. These maximum probability detections play an important role in determining which detections to use in downstream tasks.[9] We find that the BUA detector trained on the cleaned categories produces more high confidence predictions than a detector trained on the original noisy categories. Closer inspection shows that this difference is due to higher confidence when predicting objects in the new merged clean categories. This confirms our hypothesis that the original categories result in probability mass being split across multiple synonymous labels, and this issue is resolved by the new cleaned categories. We do not see the same behavior with random categories (Appendix 2).

These results support the hypothesis that noise and repetition in the original label set make it difficult to learn good distinguishing features between cate-

[9] In V&L pretraining, it is common to use the (10-100) most confident regions [16] detected in each image.

gories. They also imply that it is necessary to retrain the object detector on cleaner labels to fully improve its detection capabilities on downstream tasks.

Our experiments also show differences in the performance of the BUA Original model as implemented in Caffe and Pytorch, despite the fact that Pytorch is meant to be a reimplementation of the Caffe version. We will see similar behaviour in the visual grounding experiment later on, where the difference between the two implementations is more substantial.

6.2 Feature Space Analysis

In this section, we attempt to characterise the differences in feature space, given features from a model trained with the clean label (i.e. Clean) set vs the original model (i.e. Original). The features are from the ResNet-101's `pool5_flat` layer; these are the most common representation used for downstream tasks (e.g. visual grounding). For each image in the VG validation set, the features corresponding to the bounding box proposals are extracted. We test two confidence thresholds: with th=0.05, the models return approximately 280,000 bounding box feature vectors, whereas with th=0.2, we only evaluate approximately 100,000 features. (Different models return slightly different but comparable numbers of proposals.)

In order to be useful for downstream tasks, we expect that bounding boxes that contain similar objects should have similar features and the same predicted categories. We test this using nearest neighbors and cluster analyses.

Nearest Neighbors The local structure of the feature space can be examined using a nearest neighbors analysis: for each point in the embedding space (i.e. bounding box features), we calculate the proportion of K (with $K = 1$, 5, and 10) nearest neighbors that share the same category. This analysis is not affected by the different number of labels in the several sets and therefore it allows us to fairly compare models' embedding spaces. We expect the embedding space of the model trained with cleaner categories to be clustered better than the other embedding spaces. In other words, we expect that each point has more neighbors that share the same category when using cleaned labels.

Table 2 reports the results of this analysis, considering features extracted with different threshold values (i.e. 0.05 and 0.2) and considering either all features or only features from different images ("Filtered Neighbors"). This step removes features that might be from highly overlapping regions of the same image.

Overall, as expected, the bounding boxes extracted by the model trained on the cleaned label set have higher proportions of nearest neighbors that share the same category. This difference is substantial and consistent across different values of K, thresholds. Table 3 shows that the improvement is due to better neighborhoods of features with merged labels, and only in some case better features of unmerged, original labels.

The random features (i.e. Random) present results very similar to those obtained with the Original features, but with a small improvement. Surprisingly, this improvement is most evident for features of categories that are the same

Table 2: Proportion of K-nearest neighbors that share the same predicted category. Results were obtained with the models trained on the original, the random, and the clean categories. Overall, at each value of K, the embedding space of the model trained on clean categories is better clustered than those of models trained on the original and random labels.

		All Neighbors (%)			**Filtered Neighbors (%)**		
K	**Th.**	**Original**	**Random**	**Clean**	**Original**	**Random**	**Clean**
1	0.05	12.15±12.25	12.36±11.15	17.30±14.79	37.32±15.07	37.83±12.32	42.34±15.82
5	0.05	24.33±13.38	24.91±12.01	29.74±15.10	34.16±13.78	34.68±12.24	39.09±15.09
10	0.05	27.76±13.23	28.37±11.87	32.96±14.85	32.91±13.71	33.48±12.19	37.84±15.11
1	0.2	51.02±22.74	51.88±20.91	55.36±20.03	69.22±18.99	70.03±16.76	71.96±17.54
5	0.2	60.40±19.75	61.47±17.84	63.92±19.00	65.12±19.68	66.12±17.54	68.29±18.58
10	0.2	60.55±20.18	61.71±18.20	64.16±19.31	62.95±20.43	64.05±18.34	66.32±19.39

between Original and Random (Appendix 3), rather than the categories that were merged in Random, suggesting that there is an advantage to training on fewer labels overall.

Surprisingly, when features from the same image are ignored (Filtered Neighbors), the percentage of neighbors who share the same category increases dramatically. This indicates that BUA features tend to place visually similar regions (from the same image) close together, regardless of their semantic content (their predicted object label).

In conclusion, the analysis on the neighbors verified our main claim: when the BUA object detector is trained with the original noisy labels, it results in a sub-optimal representational space that can be improved simply by retraining the model on cleaner labels set.

Distances We examine the global structure of the feature space by looking at the distances between items with the same label (intra-category) and the distances between the category centroids (inter-category). If the feature space is organised by categories, then intra-category distances should be small, while inter-category distances should be larger.

Table 4 reports the inter and intra-categyr distances for features from the models trained with the original, clean, and random labels. Intra-category distance is the average Euclidean distance between features with the same predicted label, while inter-category distance is the average Euclidean distance between the centroids of each category (all averages are macro-averages over categories). We see that the Clean labels lead to categories that are clustered more closely together, evident in a lower average intra-category distance, compared to both the Original and Random labels. Counter to our hypothesis, inter-category distance is lower when using Clean labels, especially compared to the Original labels, and also slightly lower than Random labels. This indicates that the global fea-

Table 3: Proportion of K-nearest neighbors that share the same predicted category, comparing models trained using the original versus the clean categories. (See Table 6 for a comparison with random categories.) "Th." indicates the threshold values adopted for bounding box extraction. "Merged" refers to original categories that are merged into one new clean category. "Untouched" refers to those categories not merged with others during the cleaning process, and "All" refers to all the categories. Overall, the clean features are better clustered than the original features.

			All Neighbors (%)		**Filtered Neighbors (%)**	
Th.	**K**	**Categories**	**Original**	**Clean**	**Original**	**Clean**
0.05	1	All	12.15±12.25	17.30±14.79	37.32±15.07	42.34±15.82
0.05	1	Untouched	9.19±9.47	8.56±8.84	32.20±16.21	32.86±15.18
0.05	1	Merged	12.71±12.62	19.03±15.12	38.28±14.65	44.22±15.26
0.05	5	All	24.33±13.38	29.74±15.10	34.16±13.78	39.09±15.09
0.05	5	Untouched	19.71±12.27	20.35±11.77	28.62±24.39	29.48±13.68
0.05	5	Merged	25.19±13.40	31.60±14.99	35.19±13.41	40.99±14.63
0.05	10	All	27.76±13.23	32.96±14.85	32.91±13.71	37.84±15.11
0.05	10	Untouched	22.55±12.64	23.33±12.26	26.97±14.15	27.95±13.58
0.05	10	Merged	28.73±13.12	34.87±14.57	34.01±13.34	39.80±14.62
0.2	1	All	51.02±22.74	55.36±22.03	69.22±18.99	71.96±17.54
0.2	1	Untouched	43.34±21.95	41.37±21.45	62.26±23.11	60.92±22.20
0.2	1	Merged	52.14±22.64	57.29±21.40	70.23±18.09	73.48±16.22
0.2	5	All	60.40±19.75	63.92±19.00	65.12±19.68	68.29±18.58
0.2	5	Untouched	51.88±21.68	50.58±20.88	56.33±23.38	55.51±22.08
0.2	5	Merged	61.64±19.14	65.75±17.97	66.40±18.74	70.05±17.32
0.2	10	All	60.55±20.18	64.16±19.31	62.95±20.43	66.32±19.39
0.2	10	Untouched	50.83±22.63	49.92±21.42	52.89±23.80	52.12±22.38
0.2	10	Merged	61.97±19.39	66.12±18.15	64.42±19.46	68.28±18.09

ture space is also more compact overall. Surprisingly, across all feature spaces (Original, Clean, and Random) the intra-category distances are higher than the inter-category distances, suggesting that features from different categories are highly intermingled.

In order to control for label set and category size, we map the original features to the clean (i.e. "Orig.→Clean-878") or random (i.e. "Orig.→Random-878") set of categories, ensuring the same number of points in each label category, as well as the same number of labels. This results in a higher intra-category average distance, compared to the original categories, which indicates that features from merged labels are not mapped to nearby parts of the space. Notably, the clean mapping leads to only very slightly lower intra-category distances compared to the random mapping.

Table 4: Intra-category (average pairwise of points with the same label) and inter-category (average distance between categegory/label centroid) Euclidean distances in different feature spaces. Results were obtained with the models trained on original (i.e Orig.), clean, and random label sets. The model trained on cleaner labels presents lower distances in both the intra-categories and the inter-categories analysis.

Analysis	**Orig.**	**Orig.→Clean-878**	**Clean**	**Orig.→Random-878**	**Random**
Intra-Category	49.69 ±8.64	52.10 ±8.10	45.37 ±6.98	52.96 ±8.63	47.77 ±7.87
Inter-Category	47.97 ±5.31	NA	39.76 ±4.94	NA	40.19 ±5.87

Overall, our analysis of the local neighborhoods shows a positive effect of the clean label set, with more neighbors with the same label. However, the analysis of the global feature space suggests that the BUA features are not well separated according to object semantics, regardless of the label set used.

6.3 Visual Grounding Results

In this section, we investigate the utility of the features extracted with the BUA model in a visual grounding task, namely Referring Expression Comprehension, on the Flickr30K Entities dataset. Our expectation is that features extracted with the models trained on the new categories will be more coherent and useful than those extracted with the model trained on the original set of categories, leading to better performance on this downstream task.

As our visual grounding model, we use the Bilinear Attention Network [23] (BAN) model, which, even if no longer state of the art, obtains relatively good results on the Flickr30k Entities dataset. The advantage of using the BAN model is that it is a simple model that uses a straightforward fusion component to merge the text and visual information, and that requires only the Flickr30k Entities dataset for training (other models that achieve higher scores are pre-trained on much larger data sets and have more complex architecture [22, 65, 14, 28, 60]). BAN implements a simple architecture that uses only the 2048-dimensional bounding box features extracted from the object detector as the visual input features; it does not use the label predicted from the features. On the text side, the model initializes each word with its GloVe [37] embedding and uses a GRU [8] to generate a representation for the sentence. The visual and textual representations are then fused together through a bilinear attention networks. The simple fusion component allows us to see the effect of different visual feature spaces more clearly. We use the code provided by the authors[10], and no hyper-parameters were changed from the original model. The experiments were performed using an A5000 24GB GPU.

[10] https://github.com/jnhwkim/ban-vqa

Table 5: Visual Grounding results obtained with the Bilinear Attention Networks (BAN) [23] model on the Flickr30k Entities dataset. "R@K" refers to the Recall metric with the top K predictions, while "UB" refers to the upper bound results that can be achieved with the bounding boxes extracted with the indicated threshold. The features extracted with the model trained on the clean labels set consistently perform better than the original features.

		Test Set (%)				N. Bounding Boxes		
Features	**Threshold**	**R@1↑**	**R@5↑**	**R@10↑**	**UB↑**	**Min**	**Max**	**Test**
[23]	0.2	69.80	84.22	86.35	87.45	10	91	30 034
Original	0.2	73.32	84.21	85.67	86.53	2	89	20 916
Clean	0.2	73.41	85.08	86.52	87.31	2	93	21 923
Original	0.1	74.72	86.06	88.71	90.70	5	100	36 792
Clean	0.1	75.43	86.76	89.56	91.22	7	100	36 719
Original	0.05	75.41	85.46	88.86	92.38	12	100	59 256
Clean	0.05	75.75	85.88	89.52	92.67	11	100	56 731

Table 5 reports the results obtained in the visual grounding task by the BAN model trained using the features extracted by both the models trained on the original (i.e. Original) and new cleaner (i.e. Clean) label sets. Whenever BAN is trained using the Clean features, the performance of the model increases compared to the BAN model trained on the Original features. The improvement is small but consistent across bounding box thresholds and recall levels.

We also see that the BUA PyTorch implementation of the BAN model always achieves better performance than the Caffe implementation, even with fewer bounding boxes. This result implies that the implementation code used to train the object detector strongly impacts the results of the visual grounding task, although, in the object detection task, there is only a small improvement[11].

In conclusion, the results obtained with the BAN model on the visual grounding task suggest that the BUA model trained using a cleaner set of labels presents not only a well-clustered embedding space but also a more useful features representations able to improve downstream tasks.

7 Conclusion and Future Work

This paper introduced a new set of 878 category labels to retrain the BUA model, which refines the originally noisy 1600 categories by merging labels that are synonymous or have highly related meanings. We investigated the effect of using the

[11] The extracted features used in the BAN paper are not made available by the authors. However, some 'reproducibility' features (slightly different) were made available by third users (https://github.com/jnhwkim/ban-vqa/issues/44) who successfully reproduced the main paper results.

cleaner label set in terms of performance on the original object detection task, showing that the model trained on the new set of labels improves its object detection capabilities. We also analyzed the embedding space in the object detector trained on the cleaned categories and showed that it is better clustered than the embedding space derived from the original categories. Finally, we evaluated the utility of the new model as black-box feature extractor for a downstream visual-textual grounding task with the Bilinear Attention Network model. The results show that features from the new object detector can consistently improve the BAN model across commonly used object detection thresholds.

Future work involves studying the effect of using the improved label set on large pretrained language-and-vision models, such as VILBERT [33] and LXMERT [48]. Since these models use the bounding box category labels predicted by the object detector in their loss function, in addition to using the features as their visual input, removing label noise should benefit these models.

In this work, we merged the noisy categories using a skilled human annotator, which may have introduced some unwanted human bias or error into the cleaning process. Nevertheless, our approach highlights the advantage of using improved label sets, both for core object detection and downstream multimodal task performance. Future work could generate alternative cleaned categories by merging similar ones, e.g using a framework similar to Confidence Learning [36].

Acknowledgements

This work was supported in part by the Pioneer Centre for AI, DNRF grant number P1. Davide Rigoni was supported by a STSM Grant from Multi-task, Multilingual, Multi-modal Language Generation COST Action CA18231. We acknowledge EuroHPC Joint Undertaking for awarding us access to Vega at IZUM, Slovenia.

Appendix 1: Frequencies by Categories

We introduced both the set of clean and random categories deriving from the original ones. The original label set is defined by 1600 categories, while both the new clean and the random sets are defined by 878 categories. Figure 3 shows frequencies of objects appearing in the Visual Genome training split, where objects are either labeled according to the original label set (in blue), the new cleaned label set (in orange), or the random label set (in brown). The new label sets lead mostly to the removal of many low-frequency categories in the long tail, rather than creating new very frequent categories. Surprisingly, the random procedure that generated the random label set also removed the long tail of low-frequencies categories.

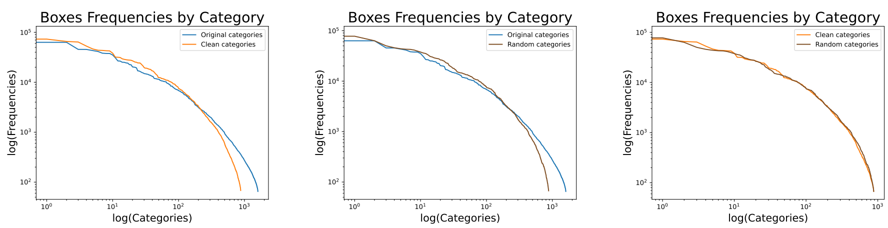

Fig. 3: LogLog plots of objects frequencies for each category. The frequencies are calculated on the training set annotations. The distribution of the original categories is in blue, the new categories are in orange, and the random categories are in brown. The cleaning process did not generate high-frequency categories and at the same time removed many low-frequency categories for both cleaner and random label sets.

Appendix 2: Prediction Confidence

In Figure 4 it is reported the KDE plots for the probability values of the argmax category predicted by the original, clean, and random label sets.

We find that the BUA detector trained on the cleaned categories produces more high confidence predictions than a detector trained on the original noisy categories. Closer inspection shows that this difference is due to higher confidence when predicting objects in the new merged clean categories. However, this is not the case for BUA trained on random categories, which presents the same confidence as the model trained on the original categories.

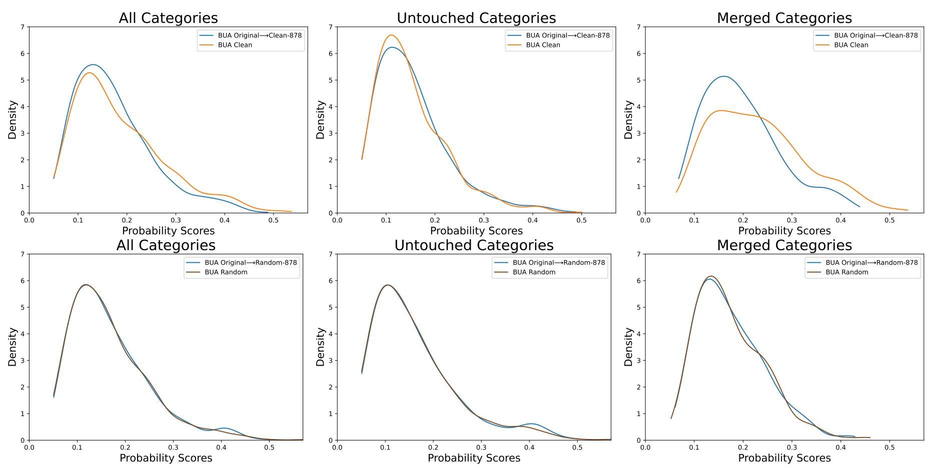

Fig. 4: KDE plots for the probability values of the argmax category predicted by the model. The plots on the left consider all the categories, the plots in the center consider just the categories that we did not merge during the cleanup process (i.e. "Untouched"), and the last plots on the right consider only the merged categories. Overall, the cleaned categories lead to higher confidence values than the original categories, while there is no difference between original and random categories.

Appendix 3: Nearest Neighbors Analysis on Random Labels

In this section, we perform the nearest neighbors analysis on the random labels focusing on the "Merged", "Untouched", and "All" categories. Table 6 reports the results of this analysis, considering features extracted with different threshold values (i.e. 0.05 and 0.2) and considering either all features or only features from different images ("Filtered Neighbors"). This step removes features that might be from highly overlapping regions of the same image.

The random features present results very similar to those obtained with the original features, but with a small improvement. In other words, there is an advantage to training on fewer labels overall. However, the improvement given by clean labels is much greater than that obtained with the random labels, strengthening the importance of training BUA with clean categories.

Table 6: Proportion of K-nearest neighbors that share the same predicted category, comparing models trained using the original versus random categories (cf. Table 3). The random features present small improvements over the original features, suggesting that there is a small advantage in training with fewer labels; however clean labels help more.

			All Neighbors (%)		**Filtered Neighbors (%)**	
Th.	**K**	**Categories**	**Original**	**Random**	**Original**	**Random**
0.05	1	All	12.15±12.25	12.36±11.15	37.32±15.07	37.83±12.32
0.05	1	Untouched	10.06±11.91	10.32±12.13	35.81±13.91	36.33±14.03
0.05	1	Merged	13.16±12.29	11.35±10.50	38.05±15.55	38.56±12.90
0.05	5	All	24.33±24.38	24.91±12.01	34.16±13.78	34.68±12.24
0.05	5	Untouched	22.66±12.60	23.12±12.61	33.09±12.88	33.54±12.78
0.05	5	Merged	25.13±13.66	25.77±11.61	34.68±14.16	35.23±11.93
0.05	10	All	27.76±13.23	28.37±11.87	32.91±13.71	33.48±12.19
0.05	10	Untouched	26.40±12.34	26.98±12.04	31.89±12.78	32.42±12.71
0.05	10	Merged	28.42±13.60	29.04±11.55	33.39±14.12	33.99±11.89
0.2	1	All	51.02±22.74	51.88±20.91	69.22±18.99	70.03±16.76
0.2	1	Untouched	45.05±21.50	46.30±21.68	65.93±17.39	66.70±17.10
0.2	1	Merged	53.84±22.73	54.70±19.93	70.98±19.37	71.72±16.33
0.2	5	All	60.40±19.75	61.47±17.84	65.12±19.68	66.12±17.54
0.2	5	Untouched	56.60±18.22	57.61±17.99	61.87±18.10	62.75±17.67
0.2	5	Merged	62.33±20.20	63.42±17.45	66.79±20.22	67.82±17.23
0.2	10	All	60.55±20.18	61.71±18.20	62.95±20.43	64.05±18.34
0.2	10	Untouched	57.05±18.44	58.14±18.24	59.76±18.69	60.69±18.39
0.2	10	Merged	62.31±20.78	63.51±17.91	64.56±21.06	65.75±18.07

Appendix 4: Clean Labels

The cleaning process produces a new set of 878 categories from the original 1600 categories, which we report below.

```
1:yolk,525:egg,324:eggs
2:goal
3:bathroom,1574:restroom
4:macaroni
6:toothpick
10:parrot
11:tail fin,1468:fin
13:calculator
15:toilet,85:toilet seat,302:toilet tank,385:toilet bowl,444:toilet lid
16:batter,5:umpire,14:catcher,474:baseball player,1210:baseball players,794:players
    ,92:player,78:tennis player,377:soccer player,207:pitcher
1254:referee
17:stop sign,17:stopsign,1437:sign post,941:traffic sign,589:street sign,817:signs
    ,129:sign,245:stop
1474:bus stop
18:cone,576:cones,560:traffic cone,658:safety cone
19:microwave,19:microwave oven
20:skateboard ramp
21:tea
23:products
25:kettle,67:tea kettle
26:kitchen
27:refrigerator,27:fridge
28:ostrich
29:bathtub,196:bath tub,306:tub
1168:blind,30:blinds
31:court,39:tennis court
314:urinals,32:urinal
34:bed,893:beds,947:bedding,660:bedspread,1343:bed frame
35:flamingo
36:giraffe,38:giraffes,471:giraffe head
37:helmet
1229:laptops,41:laptop,1124:laptop computer
42:tea pot,562:teapot
43:horse,187:horses,1319:pony
44:television,44:tv
1351:short,45:shorts
46:manhole,1014:manhole cover
47:dishwasher,148:washer
49:sail
125:parasail,1569:parachute
51:man,1511:young man,683:men,774:guy,1441:male
52:shirt,1404:tshirt,1404:t shirt,1404:t-shirt,1226:dress shirt,1099:tee shirt,1157:
    sweatshirt,653:undershirt,233:tank top,133:jersey,1288:blouse
686:cars,53:car,955:passenger car,1334:sedan
1479:police car
54:cat,185:cats,477:kitten,1117:kitty
55:garage door
56:bus,380:buses
57:radiator,1006:heater
58:tights
60:racket,60:racquet
251:home plate
1362:home
895:base
61:plate,956:plates,1378:paper plate,540:saucer,587:dishes,788:dish
65:ocean,1214:sea
63:beach
327:sand
1587:shoreline,816:shore
64:trolley
66:headboard,66:head board
68:wetsuit,217:wet suit
69:tennis racket,69:tennis racquet
70:sink,692:sinks,1123:bathroom sink,1424:basin
815:trains,71:train,1448:passenger train,899:train front,626:train car,1182:train
    cars,490:carriage,637:locomotive,1275:caboose,1318:railroad
73:sky,1217:weather
1273:skies
75:train station,272:train platform,319:platform,387:station
76:stereo
77:bats,301:bat,657:baseball bat
79:toilet brush
80:lighter
83:hair dryer
142:elephants,84:elephant
86:zebra,88:zebras
87:skateboard,87:skate board,1224:skateboards
89:floor lamp,1426:table lamp,1083:lamps,225:lamp,161:chandelier,905:light fixture
91:woman,749:women,858:lady,996:she,1486:ladies,1245:mother,1539:bride
93:tower
685:bicycles,94:bicycle,506:bikes,100:bike
95:magazines,1096:magazine
```

```
96:christmas tree
495:umbrellas,97:umbrella,1523:parasol
151:cows,98:cow,428:bull,793:cattle,583:ox,1202:calf
280:herd
99:pants,1492:pant,781:trouser,1111:sweatpants,973:jean,48:jeans,651:snow pants,503:
    ski pants,1344:slacks
102:living room
103:latch
104:bedroom
1204:grapes,105:grape
106:castle
107:table,1301:tables,875:end table,200:coffee table
108:swan
109:blender
110:orange,408:oranges
219:teddy bears,111:teddy bear,1293:teddy,270:stuffed animals,767:stuffed animal,647:
    stuffed bear
113:meter,1481:meters,211:parking meter
115:runway
262:ski boots,117:ski boot
118:dog,338:dogs,1532:puppy
119:clock,1393:clocks,7:alarm clock,1274:clock hand,509:clock face
1023:hour hand
120:hair,505:mane,1187:bangs
121:avocado
123:skirt
124:frisbee
126:desk
128:mouse,486:computer mouse
134:reigns,574:bridle,24:halter,1388:harness
1321:hot dogs,1321:hotdogs,135:hot dog,135:hotdog,1384:sausage
136:surfboard,136:surf board,351:surfboards
163:glasses,138:glass
1493:wine glasses,614:wine glass
625:sunglasses,990:eye glasses,800:eyeglasses
1327:shades,620:shade
1139:snow board,139:snowboard
140:girl,754:girls,953:little girl
141:plane,532:planes,489:airplanes,132:airplane,536:aircraft,803:jets,545:jet
143:oven,679:oven door,198:stove
1233:range
146:area rug,335:rug,467:carpet
344:bears,147:bear,131:polar bear,283:cub
149:date
150:bow tie,578:necktie,655:neck tie,268:tie
152:fire extinguisher
153:bamboo
154:wallet
156:truck,839:trucks
158:boat,234:boats,59:sailboat,59:sail boat,421:ship,719:yacht,988:canoe,1143:kayak
159:tablet
160:ceiling
162:sheep,164:ram,231:lamb
705:kites,165:kite
166:salad,868:lettuce,1398:greens
167:pillow,332:pillows,842:pillow case,675:throw pillow
168:fire hydrant,168:hydrant
169:mug,232:cup,850:coffee cup
170:tarmac,1495:asphalt,831:pavement
171:computer,1032:computers,1053:cpu
172:swimsuit,1174:swim trunks,388:bikini,1008:bathing suit
173:tomato,665:tomatoes,426:tomato slice
174:tire,1456:tires
175:cauliflower
177:snow
178:building,670:buildings,581:skyscraper,1193:second floor
1581:sandwiches,179:sandwich,1052:sandwhich
180:weather vane,753:vane
181:bird,1000:birds
182:jacket,381:coat,1521:ski jacket,566:suit jacket,836:blazer
183:chair,699:chairs,552:office chair,390:lounge chair,157:beach chair,504:seat,1022:
    seats,242:stool,1015:stools,1325:recliner
184:water,1429:ocean water
186:soccer ball,1235:balls,568:ball,481:tennis ball,674:baseball
189:barn
190:engine,619:engines,567:train engine,1093:jet engine
191:cake,12:birthday cake,273:cupcake,764:frosting
192:head
193:head band,368:headband
780:skiers,194:skier,1009:skiier
195:town
197:bowl,1027:bowls
199:tongue
1241:floors,201:floor,1556:tile floor,1310:flooring
519:uniforms,202:uniform
203:ottoman,424:sofa,137:couch,228:armchair
204:broccoli
205:olive,1148:olives
206:mound,459:pitcher's mound
1530:jug
```

```
208:food ,703:meal
209:paintings ,346:painting
210:traffic light ,1347:traffic lights
212:bananas ,531:banana ,554:banana peel ,464:banana bunch ,266:banana slice
958:peel
213:mountain ,457:mountains ,1304:mountain top ,984:mountain range ,1487:peak ,1375:
    mountainside
1161:landscape
214:cage
218:radish
221:suitcase ,221:suit case ,429:suitcases ,297:luggage
507:drawer ,222:drawers
1069:grasses ,223:grass ,488:lawn ,963:turf
101:field ,1286:grass field ,418:pasture
667:soccer field ,114:baseball field ,763:infield ,729:outfield ,22:dugout
289:apples ,224:apple
226:goggles ,1246:ski goggles
510:boys ,227:boy
229:ramp
269:burners ,230:burner
235:hat ,798:cowboy hat ,487:cap ,721:baseball cap ,1153:beanie ,1149:ball cap
922:brim
239:visor
236:soup
238:necklace
240:coffee
241:bottle ,379:bottles ,1554:beer bottle ,931:wine bottle ,476:water bottle
1267:surfers ,244:surfer
1203:back pack ,246:backpack
1498:pack
247:shin guard ,876:shin guards
248:wii remote ,432:remotes ,805:remote ,348:remote control ,723:controller ,812:game
    controller ,1208:controls ,1589:control ,1303:wii
1101:walls ,249:wall ,62:rock wall ,1220:stone wall ,1279:brick wall
250:pizza slice ,127:pizza ,914:pizzas
1466:slices ,1005:slice
252:van ,1281:minivan ,669:suv ,704:station wagon
253:packet
1402:earring ,254:earrings
255:wristband ,569:wrist band
797:track ,256:tracks
257:mitt ,1256:baseball mitt ,1454:catcher 's mitt ,1049:baseball glove
258:dome
259:snowboarder
260:faucet ,1328:tap
261:toiletries
263:room
806:snowsuit ,265:snow suit
591:benches ,267:bench ,1191:park bench
271:zoo
717:curtains ,274:curtain ,872:drape ,188:drapes
275:ear ,524:ears
276:tissue box ,1198:tissues ,1519:tissue
277:bread ,384:bun
792:toast
329:scissor ,278:scissors
412:vases ,279:vase
281:smoke
284:tail ,443:tails
285:cutting board
286:wave ,713:waves ,1311:surf
288:windshield
290:mirror ,1363:side mirror
291:license plate ,1541:license
382:trees ,292:tree ,1185:pine trees ,688:pine tree ,1436:tree line
1562:tree branch ,1356:tree branches ,933:tree trunk
1575:twig ,1271:twigs
999:branches ,1067:branch
833:wheels ,293:wheel ,791:front wheel ,666:back wheel
294:ski pole ,890:ski poles
295:clock tower
296:freezer
299:mousepad ,1257:mouse pad
300:road ,584:roadway ,122:highway ,1056:dirt road ,309:street ,353:lane ,1137:intersection
304:neck
305:cliff
307:sprinkles
308:dresser ,303:vanity
310:wing ,1232:wings ,145:tail wing
311:suit
761:outfit
312:veggie ,861:veggies
460:palm tree ,313:palm trees
1040:doors ,315:door ,1490:glass door
316:propeller
317:keys ,840:key
411:skatepark ,318:skate park
320:pot ,1551:pots
321:towel ,363:towels ,1195:hand towel
322:computer monitor ,220:monitors ,50:monitor ,597:computer screen ,116:screen
```

```
1199:flip flops,323:flip flop,1077:sandal,1176:sandals
325:shed
328:face
500:cart,330:carts
331:squash,515:pumpkin
334:glove,1298:gloves
336:watch,1196:wristwatch,1555:wrist watch
337:grafitti
339:scoreboard
340:basket,1500:baskets
341:poster
342:duck,352:ducks
343:horns,527:horn
345:jeep
347:lighthouse
349:toaster
1166:vegetable,350:vegetables,784:produce
354:carrots,530:carrot
355:market
659:paper towel,356:paper towels
357:island
358:blueberries,1533:berries,1462:strawberries,1061:strawberry,888:blueberry
359:smile
360:balloons,416:balloon
361:stroller
594:napkins,362:napkin
915:paper,364:papers
365:person,635:adult,949:worker,943:pedestrian
541:people,461:crowd,795:group,940:audience,1197:spectator,615:spectators,1152:fans
333:family
894:fan,8:ceiling fan
1251:train track,366:train tracks
986:rail,1406:rails
367:child
369:pool
370:plant,919:plants
1382:weeds
371:harbor,643:marina
372:counter
373:hand,783:hands
374:house,978:houses
375:donut,375:doughnut,628:donuts,628:doughnuts
376:knot
378:seagull
386:trunk,1140:trunks
391:breakfast
392:nose,491:snout,668:nostril
393:moon
394:river,1588:stream
395:racer
1103:pictures,396:picture,1529:image,1070:photo,9:photos,1453:photograph
397:shaker,804:shakers,81:pepper shaker,991:salt shaker,1573:salt,1522:seasoning
1542:peppers,623:pepper
398:sidewalk,398:side walk
907:curb
399:shutters,1004:shutter
400:stove top,400:stovetop
401:church,472:steeple,1126:spire
402:lampshade,687:lamp shade
403:map
406:airport
410:enclosure
413:city
414:park
415:mailbox
417:billboard,631:advertisement,1211:ad
419:portrait
420:forehead
422:cookie
423:seaweed
425:slats
427:tractor
430:graffiti
837:pen,433:pens
1415:windowsill,434:window sill,1284:ledge
435:suspenders
436:easel
437:tray,405:platter
438:straw
439:collar
440:shower,130:shower curtain,965:shower head,997:shower door
864:bags,441:bag,1158:handbag,728:purse,821:sack
445:panda
447:outlet,1455:electrical outlet,1434:socket,592:fuselage
1154:stem,448:stems
449:valley
450:flag,1545:flags,718:american flag
451:jockey
452:gravel
453:mouth
```

```
454:window,979:windows,1422:side window,537:front window,282:skylight,1586:panes
455:bridge
1432:overpass
456:corn
458:beer
609:ski,462:skis,1337:skiis
465:tennis shoe,1173:tennis shoes,748:sneakers,904:sneaker,771:shoes,243:shoe,520:
     cleat,1340:cleats,706:boots,873:boot
468:eye,547:eyes
469:urn
470:beak
473:mattress
475:wine
478:archway,1549:arches,636:arch
929:candles,479:candle
480:croissant
482:dress
483:column,1496:columns
1238:utensil,484:utensils,765:forks,264:fork,1179:butter knife,757:knife,557:spoon
     ,517:chopsticks,542:silverware,1316:knives
622:cellphone,485:cell phone,463:phone,813:smartphone,582:telephone,514:iphone
498:ipod
492:cabinets,611:cabinet,558:cabinet door,819:cupboards,1330:cupboard
493:lemons,678:lemon
494:grill
496:meat,1380:beef
497:wagon
499:bookshelf,863:book shelf,848:shelf,887:bookcase,1163:shelves
501:roof
502:hay
508:game
555:baseball game
74:match,638:tennis match,974:tennis
511:rider
512:fire escape
1535:pans,516:pan,1295:skillet
588:hills,518:hill,1132:hill side,1175:hillside,513:slope,1025:ski slope
521:costume
522:cabin
523:police officer,431:policeman,855:officer,826:police
1268:arrows,528:arrow\scriptsize
529:toothbrush
533:garden,768:yard
534:forest,409:woods,1228:wood
535:brocolli
538:dashboard
1222:statues,539:statue,682:monument,1332:sculpture
571:fruits,543:fruit
544:drain
546:speaker,1058:speakers
549:lid
550:soap
601:rock,551:rocks,1087:stone,967:stones,845:boulder,1457:boulders
553:door knob,976:doorknob,698:knob,607:knobs
556:asparagus
559:pineapple
561:nightstand,561:night stand
563:taxi,1265:taxi cab,901:cab
564:chimney
565:lake
865:pickles,570:pickle
572:pad,1369:pads,33:knee pads,994:knee pad,747:kneepad
575:breast
880:head light,577:headlight,590:headlights
579:skater,298:skateboarder
580:toilet paper
1160:socks,585:sock
586:paddle,1464:oar
593:card
807:bushes,595:bush,1336:shrubs,1305:shrub,287:hedges,215:hedge
596:rice
1183:spoke,598:spokes
599:flowers,663:flower,689:bouquet
600:bucket
603:pear,1491:pears
604:sauce,608:mustard,786:ketchup,1566:condiments
605:store,404:shop,1131:storefront
866:stand
610:stands,985:bleachers
612:dirt,466:ground,1272:soil,1476:pebbles,1477:mud
613:goats,712:goat
617:pancakes
673:kid,618:kids,1063:children
621:feeder
624:blanket,446:comforter,1200:quilt
627:magnet,641:magnets
629:sweater,407:hoodie,645:vest
630:signal
632:log
633:vent,1043:air vent
```

```
634:whiskers
1452:tents,639:tent
939:motor bike,640:motorbike,144:dirt bike,326:moped,442:scooter,1194:motor,40:
    motorcycle,216:motorcycles
642:night
644:wool
646:railroad tracks,548:railway
649:bib
650:frame,1019:picture frame
652:tank,734:water tank,892:gas tank
654:icons
656:beams,785:beam
661:can
1162:soda can
1565:containers,662:container
664:vehicle,1585:vehicles
671:canopy
672:flame
676:belt
677:rainbow
758:tags,680:tag,1482:name tag,1401:name
681:books,1011:book
1469:shadows,684:shadow
690:toothpaste
1094:potatoes,691:potato
693:hook
694:switch,1033:light switch
695:lamp post,695:lamppost,1520:light post
696:lapel
697:desert
700:pasta
701:feathers,1598:feather,155:tail feathers
702:hole
707:baby
708:biker,746:motorcyclist
709:gate
710:signal light,1156:traffic signal
711:headphones
714:bumper
715:bud,1201:floret
716:logo
720:box,1107:boxes,616:crate,982:cardboard box,1417:package,1116:bin,1397:carton
724:awning
725:path,778:pathway,1447:trail
730:pigeon
731:toddler
732:beard,869:facial hair,389:goatee,648:moustache,1219:mustache
735:board
736:parade
737:robe
738:newspaper
1136:wire,739:wires
740:camera
742:deck
743:watermelon,1031:melon
782:cloud,744:clouds
745:deer
1361:onion,750:onions
1512:eyebrows,751:eyebrow
752:gas station
755:trash
759:light,1261:lights
760:bunch
762:groom
766:entertainment center,1035:tv stand
770:ladder
1169:bracelets,772:bracelet
773:teeth
775:display case
1068:display
776:cushion,1407:cushions
1247:posts,777:post\scriptsize
802:table cloth,779:tablecloth
1385:paws,787:paw
789:raft
790:crosswalk
796:coffee pot
799:petal,1596:petals
801:handle,1057:handles
1017:door handle
808:dessert
830:legs,809:leg,726:front legs
810:eagle
811:fire truck,811:firetruck
814:backsplash
818:bell
820:sweat band,1365:sweatband
822:ankle
823:coin slot
824:bagel
```

```
825:masts,1046:mast
828:biscuit
1074:toys,829:toy
1346:doll
832:outside
834:driver
835:numbers,992:number
838:cabbage
841:saddle
843:goose,383:geese
844:label
846:pajamas
847:wrist
849:cross
854:air
856:pepperoni
857:cheese
859:kickstand
936:countertop,860:counter top
862:baseball uniform
867:netting,1570:mesh
112:net,883:tennis net
870:lime
884:animal,871:animals
874:railing,1475:railings
237:fence,1412:wire fence,1390:fence post,1283:fencing
1563:tusks,879:tusk
881:walkway,885:boardwalk
882:cockpit
891:parking lot,852:lot
573:dispenser,896:soap dispenser
897:banner
898:life vest,727:life jacket
1180:words,900:word,1597:text
903:exhaust pipe
1248:power line,906:power lines
908:scene
909:buttons,1089:button
910:roman numerals,960:roman numeral,769:numeral,756:numerals
911:muzzle
912:sticker,1170:stickers,1387:decal
913:bacon
917:stairs,877:steps,1484:staircase,1423:stairway
918:triangle
921:beans,1135:bean
924:letters,1472:letter,1122:lettering
926:menu
983:fingers,927:finger,733:thumb
930:picnic table
932:pencil
934:nail
935:mantle
176:fireplace
937:view
938:line,1155:lines,1560:baseline
1467:arms,942:arm
944:stabilizer
945:dock,1138:pier
946:doorway
950:canal
951:crane
952:grate
954:rims,1066:rim
957:background
1349:strings,961:string
920:rope
1297:cable
1165:cord,1528:cords
962:tines
964:armrest
966:leash
1147:stop light,968:stoplight
970:front
948:end
971:scarf
972:band
975:pile,1192:stack
977:foot,916:feet
980:restaurant
981:booth
987:pastry,741:pastries
989:sun,1002:sunset
993:fish
995:fur
998:rod
1001:printer
1003:median
1007:prongs
1010:rack
1012:blade,1592:blades
```

```
1013:apartment
1016:overhang
1018:couple
1020:chicken
1021:planter
1024:dvd player
1026:french fry,722:fries,90:french fries,853:fry
1028:top
1029:landing gear
1030:coffee maker
1034:jar
1036:overalls
1037:garage
1038:tabletop
1039:writing
1041:stadium
1042:placemat
1044:trick
1045:sled
1047:pond
1048:steering wheel
1050:watermark,1580:print,1141:website
1051:pie
1064:crust
1054:mushroom,1461:mushrooms
1059:fender
1060:telephone pole,1055:power pole,1367:light pole,1292:utility pole
1090:poles,602:pole
1062:mask,1112:face mask
1065:art,1371:artwork,827:drawing
1071:receipt
1072:instructions
1073:herbs
1366:handlebar,1075:handlebars,969:handle bars
1076:trailer
1078:skull
1079:hangar
1080:pipe,1416:pipes
1081:office
1082:chest
1084:horizon
1085:calendar
1086:foam
1517:bar,1088:bars
1091:heart
1092:hose
1095:rain
1097:chain
1098:footboard,1553:baseboard
1100:design,1451:designs
1102:copyright
1584:pillars,1104:pillar
1266:drinks,1105:drink,606:juice,923:beverage,925:soda,902:liquid
1106:barrier
1108:chocolate
1109:chef
1110:slot
1113:icing
1115:circle
1118:electronics,1567:device
1119:wild
1374:tile,1120:tiles
1121:steam
1125:cherry
1127:conductor
1128:sheet,1189:sheets
1129:slab
1130:windshield wipers,1471:windshield wiper,1114:wipers
1133:spatula
1345:tail lights,1134:tail light,1134:taillight,959:brake light
1142:bolt
1144:nuts
1145:holder
1146:turbine
1151:barrel
1159:mulch
1167:apron
1171:traffic
1172:strip
1177:concrete,1544:cement
1178:lips,1444:lip
1181:leaves,851:foliage,1282:leaf
1184:cereal
1186:cooler
1188:half
1190:figurine
1205:ski tracks
1206:skin
1209:dinner
1207:bow,1212:ribbon
```

1213: hotel
1215: cover
1216: tarp
1218: notebook
1221: closet
1223: bank
1225: butter
1227: knee
1230: cuff
1231: hubcap
1234: structure
1236: tunnel
1237: globe
1239: dumpster
1240: cd ,928: dvds ,1510: disc
1242: wrapper
1243: folder
1244: pocket
1249: wake
1294: rose ,1250: roses
1252: reflection
1253: air conditioner
1255: barricade
1258: garbage can ,1360: trash bin ,889: trash bag ,878: trashcan ,526: trash can
1259: buckle
1260: footprints
1262: muffin
1263: bracket
1264: plug
1269: control panel ,1506: panel
1270: ring
1276: playground
1277: mango
1278: stump
1280: screw
1312: cloth
1285: clothes ,1342: clothing
1287: plumbing
1289: patch
1290: scaffolding
1291: hamburger ,1150: burger
1296: cycle
1299: bark
1300: decoration
1302: palm
1306: hoof
1307: celery
1308: beads
1309: plaque
1540: spray
1543: passengers ,1313: passenger
1314: spot ,1599: spots
1315: plastic
1317: case
1320: muffler
1322: stripe ,1392: stripes
1323: scale
1324: block ,1503: blocks
1326: body
1329: tools
1331: wallpaper
1333: surface
1335: distance
1338: lift ,1164: ski lift
1339: bottom
1341: roll
1348: symbol
1350: fixtures
1352: paint
1353: candle holder
1354: guard rail
1355: cyclist
1357: ripples
1358: gear
1359: waist
1364: brush
1370: ham
1372: reflector
1373: figure
1376: black
1409: brick ,1377: bricks
1379: stick
1381: patio
82: gazebo
1383: back
1386: farm
1389: monkey
1391: door frame
1428: pony tail ,1394: ponytail
1395: toppings

1396:strap
1399:chin
1400:lunch
1403:area
1405:cream
1408:lanyard
1410:hallway
1411:cucumber
1413:fern
1414:tangerine
1418:wheelchair
1419:chips
1420:driveway
1421:tattoo
1425:machine
1427:radio
1430:inside
1431:cargo
1433:mat
1435:flower pot
1438:tube
1439:dial
1440:splash
1442:lantern
1443:lipstick
1445:tongs
1446:ski suit
1449:bandana
1450:antelope
1458:mannequin
1459:plain
1460:layer
1463:piece
1465:bike rack
1470:hood
1473:dot
1478:claws
1480:crown
1483:entrance
1485:shrimp
1488:vines,1577:ivy
1489:computer keyboard,886:keypad,72:keyboard
1494:stall
1497:sleeve
1499:cheek
1501:land
1502:day
1504:courtyard
1505:pedal
1507:seeds
1508:balcony
1509:yellow
1513:crumbs
1514:spinach
1515:emblem
1516:object,1548:objects
1518:cardboard
1524:terminal
1525:surfing
1526:streetlight,1526:street light,1368:street lamp
1527:alley
1531:antenna
1534:diamond
1536:fountain
1537:foreground
1538:syrup
1546:shack,1571:hut
1547:trough
1550:streamer
1552:border
1557:page
1558:pin
1559:items
1564:donkey
1568:envelope
1572:butterfly
1576:pilot
1578:furniture
1579:clay
1582:lion
1583:shingles
1590:lock
1591:microphone
1593:towel rack,1561:hanger
1594:coaster
1595:star
1600:buoy

References

1. Anderson, P., He, X., Buehler, C., Teney, D., Johnson, M., Gould, S., Zhang, L.: Bottom-up and top-down attention for image captioning and visual question answering. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 6077–6086 (2018)
2. Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Zitnick, C.L., Parikh, D.: Vqa: Visual question answering. In: ICCV. pp. 2425–2433 (2015)
3. Beyer, L., Hénaff, O.J., Kolesnikov, A., Zhai, X., Oord, A.v.d.: Are we done with imagenet? https://doi.org/10.48550/ARXIV.2006.07159, https://arxiv.org/abs/2006.07159
4. Bugliarello, E., Cotterell, R., Okazaki, N., Elliott, D.: Multimodal pretraining unmasked: A meta-analysis and a unified framework of vision-and-language berts. arXiv preprint arXiv:2011.15124 (2020)
5. Cadene, R., Ben-Younes, H., Cord, M., Thome, N.: Murel: Multimodal relational reasoning for visual question answering. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 1989–1998 (2019)
6. Cadene, R., Dancette, C., Cord, M., Parikh, D., et al.: Rubi: Reducing unimodal biases for visual question answering. Advances in neural information processing systems **32** (2019)
7. Chen, K., Gao, J., Nevatia, R.: Knowledge aided consistency for weakly supervised phrase grounding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4042–4050 (2018)
8. Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078 (2014)
9. Dai, X., Chen, Y., Xiao, B., Chen, D., Liu, M., Yuan, L., Zhang, L.: Dynamic head: Unifying object detection heads with attentions. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 7373–7382 (2021)
10. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
11. Dost, S., Serafini, L., Rospocher, M., Ballan, L., Sperduti, A.: Jointly linking visual and textual entity mentions with background knowledge. In: International Conference on Applications of Natural Language to Information Systems. pp. 264–276. Springer (2020)
12. Dost, S., Serafini, L., Rospocher, M., Ballan, L., Sperduti, A.: On visual-textual-knowledge entity linking. In: ICSC. pp. 190–193. IEEE (2020)
13. Dost, S., Serafini, L., Rospocher, M., Ballan, L., Sperduti, A.: Vtkel: a resource for visual-textual-knowledge entity linking. In: ACM. pp. 2021–2028 (2020)
14. Dou, Z.Y., Kamath, A., Gan, Z., Zhang, P., Wang, J., Li, L., Liu, Z., Liu, C., LeCun, Y., Peng, N., et al.: Coarse-to-fine vision-language pre-training with fusion in the backbone. arXiv preprint arXiv:2206.07643 (2022)
15. Frank, S., Bugliarello, E., Elliott, D.: Vision-and-language or vision-for-language. On Cross-Modal Influence in Multimodal Transformers.(2021). DOI: https://doi.org/10.18653/v1/2021. emnlp-main **775** (2021)
16. Gan, Z., Li, L., Li, C., Wang, L., Liu, Z., Gao, J., et al.: Vision-language pre-training: Basics, recent advances, and future trends. Foundations and Trends® in Computer Graphics and Vision **14**(3–4), 163–352 (2022)

17. Gupta, T., Vahdat, A., Chechik, G., Yang, X., Kautz, J., Hoiem, D.: Contrastive learning for weakly supervised phrase grounding. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part III 16. pp. 752–768. Springer (2020)
18. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE (jun 2016). https://doi.org/10.1109/cvpr.2016.90
19. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016. pp. 770–778. IEEE Computer Society (2016). https://doi.org/10.1109/CVPR.2016.90, https://doi.org/10.1109/CVPR.2016.90
20. Jing, C., Jia, Y., Wu, Y., Liu, X., Wu, Q.: Maintaining reasoning consistency in compositional visual question answering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5099–5108 (2022)
21. Kafle, K., Shrestha, R., Kanan, C.: Challenges and prospects in vision and language research. Frontiers in Artificial Intelligence **2**, 28 (2019)
22. Kamath, A., Singh, M., LeCun, Y., Synnaeve, G., Misra, I., Carion, N.: MDETR - modulated detection for end-to-end multi-modal understanding. In: 2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021. pp. 1760–1770. IEEE (2021). https://doi.org/10.1109/ICCV48922.2021.00180, https://doi.org/10.1109/ICCV48922.2021.00180
23. Kim, J.H., Jun, J., Zhang, B.T.: Bilinear attention networks. Advances in neural information processing systems **31** (2018)
24. Kiros, R., Salakhutdinov, R., Zemel, R.S.: Unifying visual-semantic embeddings with multimodal neural language models. arXiv preprint arXiv:1411.2539 (2014)
25. Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L.J., Shamma, D.A., et al.: Visual genome: Connecting language and vision using crowdsourced dense image annotations. International journal of computer vision **123**(1), 32–73 (2017)
26. Li, L., Gan, Z., Cheng, Y., Liu, J.: Relation-aware graph attention network for visual question answering. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10313–10322 (2019)
27. Li, L.H., Yatskar, M., Yin, D., Hsieh, C.J., Chang, K.W.: Visualbert: A simple and performant baseline for vision and language. arXiv preprint arXiv:1908.03557 (2019)
28. Li, L.H., Zhang, P., Zhang, H., Yang, J., Li, C., Zhong, Y., Wang, L., Yuan, L., Zhang, L., Hwang, J.N., et al.: Grounded language-image pre-training. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10965–10975 (2022)
29. Li, W.H., Yang, S., Wang, Y., Song, D., Li, X.Y.: Multi-level similarity learning for image-text retrieval. Information Processing & Management **58**(1), 102432 (2021)
30. Li, X., Yin, X., Li, C., Zhang, P., Hu, X., Zhang, L., Wang, L., Hu, H., Dong, L., Wei, F., et al.: Oscar: Object-semantics aligned pre-training for vision-language tasks. In: European Conference on Computer Vision. pp. 121–137. Springer (2020)
31. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)

32. Liu, L., Ouyang, W., Wang, X., Fieguth, P., Chen, J., Liu, X., Pietikäinen, M.: Deep learning for generic object detection: A survey. International journal of computer vision **128**(2), 261–318 (2020)
33. Lu, J., Batra, D., Parikh, D., Lee, S.: Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In: Wallach, H.M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E.B., Garnett, R. (eds.) Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada. pp. 13–23 (2019), https://proceedings.neurips.cc/paper/2019/hash/c74d97b01eae257e44aa9d5bade97baf-Abstract.html
34. Mao, J., Xu, W., Yang, Y., Wang, J., Yuille, A.L.: Deep captioning with multimodal recurrent neural networks (m-rnn). In: Bengio, Y., LeCun, Y. (eds.) ICLR (2015)
35. Mogadala, A., Kalimuthu, M., Klakow, D.: Trends in integration of vision and language research: A survey of tasks, datasets, and methods. Journal of Artificial Intelligence Research **71**, 1183–1317 (2021)
36. Northcutt, C., Jiang, L., Chuang, I.: Confident learning: Estimating uncertainty in dataset labels. Journal of Artificial Intelligence Research **70**, 1373–1411 (2021)
37. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). pp. 1532–1543 (2014)
38. Redmon, J., Farhadi, A.: Yolo9000: better, faster, stronger. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 7263–7271 (2017)
39. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems **28** (2015)
40. Rigoni, D., Serafini, L., Sperduti, A.: A better loss for visual-textual grounding. In: Hong, J., Bures, M., Park, J.W., Cerný, T. (eds.) SAC '22: The 37th ACM/SIGAPP Symposium on Applied Computing, Virtual Event, April 25 - 29, 2022. pp. 49–57. ACM (2022). https://doi.org/10.1145/3477314.3507047, https://doi.org/10.1145/3477314.3507047
41. Rigoni, D., Serafini, L., Sperduti, A.: A better loss for visual-textual grounding. In: Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing. pp. 49–57 (2022)
42. Rohrbach, A., Rohrbach, M., Hu, R., Darrell, T., Schiele, B.: Grounding of textual phrases in images by reconstruction. In: European Conference on Computer Vision. pp. 817–834. Springer (2016)
43. Shih, K.J., Singh, S., Hoiem, D.: Where to look: Focus regions for visual question answering. In: CVPR. pp. 4613–4621 (2016)
44. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
45. Singh, A., Natarajan, V., Shah, M., Jiang, Y., Chen, X., Batra, D., Parikh, D., Rohrbach, M.: Towards vqa models that can read. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8317–8326 (2019)
46. Song, H., Kim, M., Park, D., Shin, Y., Lee, J.G.: Learning from noisy labels with deep neural networks: A survey. IEEE Transactions on Neural Networks and Learning Systems pp. 1–19 (2022). https://doi.org/10.1109/TNNLS.2022.3152527
47. Su, W., Zhu, X., Cao, Y., Li, B., Lu, L., Wei, F., Dai, J.: Vl-bert: Pre-training of generic visual-linguistic representations. arXiv preprint arXiv:1908.08530 (2019)

48. Tan, H., Bansal, M.: Lxmert: Learning cross-modality encoder representations from transformers. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). pp. 5100–5111 (2019)
49. Tian, Z., Shen, C., Chen, H., He, T.: Fcos: Fully convolutional one-stage object detection. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9627–9636 (2019)
50. Wang, C.Y., Bochkovskiy, A., Liao, H.Y.M.: Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. arXiv preprint arXiv:2207.02696 (2022)
51. Wang, H., Wang, H., Xu, K.: Categorizing concepts with basic level for vision-to-language. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018)
52. Wang, L., Huang, J., Li, Y., Xu, K., Yang, Z., Yu, D.: Improving weakly supervised visual grounding by contrastive knowledge distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14090–14100 (2021)
53. Wang, Q., Tan, H., Shen, S., Mahoney, M.W., Yao, Z.: Maf: Multimodal alignment framework for weakly-supervised phrase grounding. arXiv preprint arXiv:2010.05379 (2020)
54. Wang, R., Qian, Y., Feng, F., Wang, X., Jiang, H.: Co-vqa: Answering by interactive sub question sequence. In: Findings of the Association for Computational Linguistics: ACL 2022. pp. 2396–2408 (2022)
55. Wang, S., Wang, R., Yao, Z., Shan, S., Chen, X.: Cross-modal scene graph matching for relationship-aware image-text retrieval. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) (March 2020)
56. Wang, X., Zhang, S., Yu, Z., Feng, L., Zhang, W.: Scale-equalizing pyramid convolution for object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13359–13368 (2020)
57. Xu, M., Zhang, Z., Hu, H., Wang, J., Wang, L., Wei, F., Bai, X., Liu, Z.: End-to-end semi-supervised object detection with soft teacher. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3060–3069 (2021)
58. Yang, J., Li, C., Gao, J.: Focal modulation networks. arXiv preprint arXiv:2203.11926 (2022)
59. Yang, Z., Liu, S., Hu, H., Wang, L., Lin, S.: Reppoints: Point set representation for object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9657–9666 (2019)
60. Yao, Y., Chen, Q., Zhang, A., Ji, W., Liu, Z., Chua, T.S., Sun, M.: Pevl: Position-enhanced pre-training and prompt tuning for vision-language models. arXiv preprint arXiv:2205.11169 (2022)
61. Yu, Z., Yu, J., Cui, Y., Tao, D., Tian, Q.: Deep modular co-attention networks for visual question answering. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6281–6290 (2019)
62. Yu, Z., Yu, J., Xiang, C., Zhao, Z., Tian, Q., Tao, D.: Rethinking diversified and discriminative proposal generation for visual grounding. In: Proceedings of the 27th International Joint Conference on Artificial Intelligence. pp. 1114–1120 (2018)
63. Zhang, H., Niu, Y., Chang, S.F.: Grounding referring expressions in images by variational context. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4158–4166 (2018)

64. Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., Ni, L.M., Shum, H.Y.: Dino: Detr with improved denoising anchor boxes for end-to-end object detection. arXiv preprint arXiv:2203.03605 (2022)
65. Zhang, H., Zhang, P., Hu, X., Chen, Y.C., Li, L.H., Dai, X., Wang, L., Yuan, L., Hwang, J.N., Gao, J.: Glipv2: Unifying localization and vision-language understanding. arXiv preprint arXiv:2206.05836 (2022)
66. Zhang, Q., Lei, Z., Zhang, Z., Li, S.Z.: Context-aware attention network for image-text retrieval. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)
67. Zhang, S., Chi, C., Yao, Y., Lei, Z., Li, S.Z.: Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9759–9768 (2020)
68. Zhou, B., Tian, Y., Sukhbaatar, S., Szlam, A., Fergus, R.: Simple baseline for visual question answering. arXiv preprint arXiv:1512.02167 (2015)
69. Zhou, L., Palangi, H., Zhang, L., Hu, H., Corso, J., Gao, J.: Unified vision-language pre-training for image captioning and vqa. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 13041–13049 (2020)