LANGUAGE MODULATED DETECTION AND DETECTION MODULATED LANGUAGE GROUNDING IN 2D AND 3D SCENES

Anonymous authors
Paper under double-blind review

ABSTRACT

To localize an object referent, humans attend to different locations in the scene and visual cues depending on the utterance. Existing language and vision systems often model such task-driven attention using object proposal bottlenecks: a pre-trained detector proposes objects in the scene, and the model is trained to selectively process those proposals and then predict the answer without attending to the original image. Object detectors are typically trained on a fixed vocabulary of objects and attributes that is often too restrictive for open-domain language grounding, where the language utterance may refer to visual entities in various levels of abstraction, such as a cat, the leg of a cat, or the stain on the front leg of the chair. This paper proposes a model that reconciles language grounding and object detection with two main contributions: i) Architectures that exhibit iterative attention across the language stream, the pixel stream, and object detection proposals. In this way, the model learns to condition on easy-to-detect objects (e.g., "table") and language hints (e.g. "on the table") to detect harder objects (e.g., "mugs") mentioned in the utterance. ii) Optimization objectives that treat object detection as language grounding of a large predefined set of object categories. In this way, cheap object annotations are used to supervise our model, which results in performance improvements over models that are not co-trained across both referential grounding and object detection. Our model has a much lighter computational footprint, achieves faster convergence and has shown on par or higher performance compared to both detection-bottlenecked and non-detection bottlenecked language-vision models on both 2D and 3D language grounding benchmarks.

1 INTRODUCTION

Consider Figure 1. At first glance, many objects pop out, such as the chairs, the table, the shelves or the bathroom vanity. The clock on the shelf or the bottle on the vanity though do not pop out as salient, but they can be easily localised if someone draws our attention to them, e.g., by referencing them. The work of Katsuki & Constantinidis (2014) distinguishes between bottom-up attention drawn to salient parts of the scene, and top-down attention guided by the task of the agent. The goal of this work is to develop a model that combines bottom-up attention and language-driven attention for referential grounding on 2D images and 3D point clouds.

The ability to localize objects mentioned in an utterance is central for an intelligent agent to communicate, learn and be instructed by humans. As a result, there is a lot of research on models for language grounding that fuse information across image and language streams. For ease of exposition, we group existing models into two categories based on whether they pursue or not a generic, task-independent, visual tokenization of the scene:

i) **Models that tokenize the visual scene into discrete sets of entities using generic pre-trained high vocabulary object detectors** (Fang et al., 2015; Johnson et al., 2016b; Karpathy & Fei-Fei, 2017; Fukui et al., 2016; Hu et al., 2016). Upon tokenization of the visual stream, many recent approaches use large-scale transformer models to fuse information across both vision and language modalities to localize referent objects or answer questions about an image or point cloud (Lu et al., 2019; Chen et al., 2020b; Yang et al., 2021). Instead of transformer layers, neural-symbolic ap-

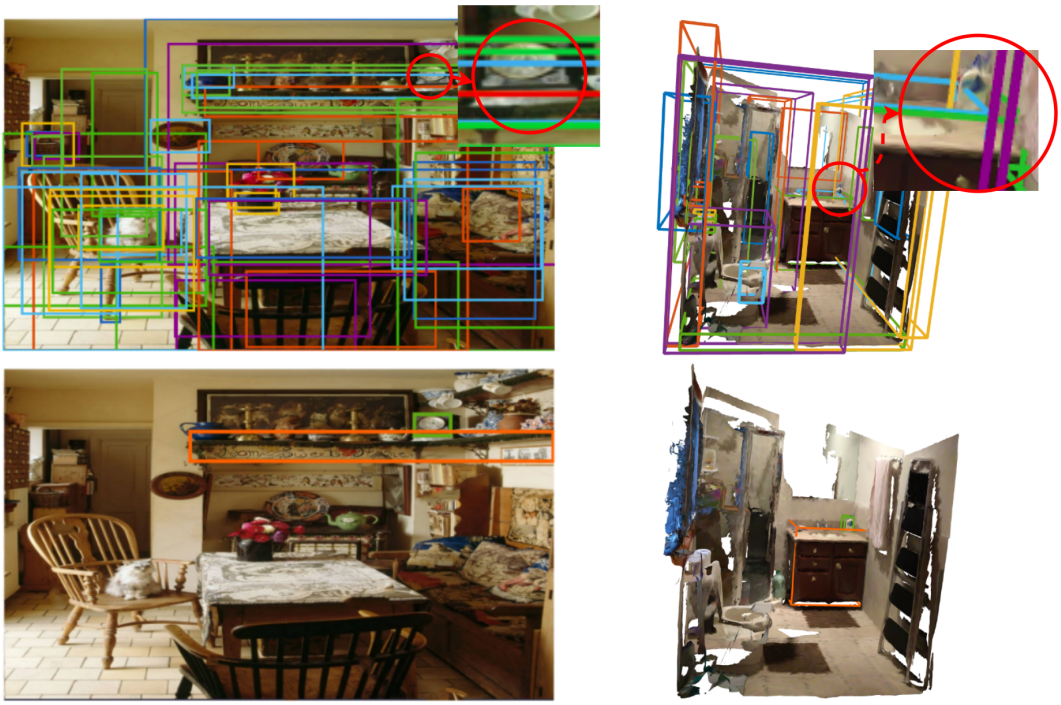


Figure 1: **Language-modulated detection for task-driven scene understanding in 2D (*left*) and 3D (*right*).** Boxes detected by object detectors (2D Faster-RCNN detector trained on 1601 Visual Genome classes and 3D Group-Free detector trained on 485 Scannet classes) often fail to localize the object of interest (clock, bottle). The proposed model locates the relevant objects by attending across image, language and box proposal streams in 2D and in 3D.

proaches (Yi et al., 2018; Mao et al., 2019) use programs of neural modules that are applied on the extracted visual tokens, their color and shape descriptors. These latter models have been mainly applied in simple domains, such as CLEVR (Johnson et al., 2016a) or CLEVRER (Yi et al., 2019), where the computational graph to answer a question or find an object is well defined and an accurate tokenization of the scene can be obtained with existing object detectors. In all of these models, the original image is discarded upon extraction of the object proposals, i.e., the visual tokens. This is problematic as a generic object detector typically fails to propose all relevant entities for the given utterance bottom-up; there are simply too many entities to be proposed and is computational infeasible to do so. Moreover, small, occluded or rare objects are hard to detect without task-driven guidance. In Figure 1, we can easily miss the clock on the shelf unless someone draws our attention to it, and indeed the state-of-the-art detector misses it. Indeed, the quality of the pre-trained detector has been found to be very crucial for the final performance of these models (Zhang et al., 2021). The particularity of the visual domain is that relevant entities come at different levels of spatial abstraction which renders task-independent tokenization hard. This is in stark contrast to the language domain where sentences are naturally tokenized into words.

ii) **Models that do not tokenize the visual scene but rather apply operations directly on pixels to extract relevant information**, either end-to-end (Lu et al., 2016; Xu et al., 2015) or using modular network architectures (Andreas et al., 2016; Hu et al., 2017; Johnson et al., 2017; Chen et al., 2021).

We propose a language grounding model for 2D and 3D scenes that uses tri-partite attention across the language stream, the pixel stream and a set of object proposals obtained by a pre-trained detector to predict the object(s) referenced in the utterance. During iterative attention, the model can propose new objects/parts, improve existing proposals , or directly produce and refine object proposals.

Our model can thus condition conditions on easy-to-detect objects, e.g. “bathroom vanity” in Figure 1, and language hints, e.g. “on top of the bathroom vanity”, to explore the scene and detect harder objects, e.g. “bottle”, mentioned in the utterance. At the same time, it converges faster be-

cause it learns to exploit already detected objects without the need to learn object detection from scratch. We call our model BEAUTY-DETR as it uses both proposals, obtained by a general purpose detector "bottom-up" , i.e., without any utterance-guided modulation, and "top-down" guidance from the input language utterance, to localize the relevant objects in the scene, and builds upon the DETR model of Carion et al. (2020) for decoding fused features into relevant object boxes. We train BEAUTY-DETR under object localization objectives and further disguise training data for object detection as language grounding training examples by providing a randomized caption comprised of a long set of category labels, from which the model should discard the irrelevant ones and localize the ones that exist in the image. This co-training scheme results in significant gains for our grounding model, especially in the 3D dataset of Achlioptas et al. (2020), where we are the first to report results without assuming oracle 3D boxes as previous work.

The closest existing work to ours is M-DETR (Kamath et al., 2021), which recently also revisited the paradigm of end-to-end attention between language and visual streams, bypassing object proposal bottlenecks by constructing a sequence of tokens from both modalities and iteratively applying attention layers. M-DETR achieves big leaps in performance at the computational cost of pre-training for many epochs on several thousands of images, which is infeasible for many users without access to large scale compute. Our model in this work i) uses direct attention on language, scene and object proposals, therefore it can quickly pick existing proposals, refine them or add more, ii) employs deformable attention (Zhu et al., 2021) in the visual stream, leading to a much lighter computational footprint when it is trained on large 2D datasets, iii) exploits supervision from object detection annotations, alongside language grounding annotations, iv) is applied to both 2D and 3D visual input, with state-of-the-art performance on 3D benchmarks (Achlioptas et al., 2020; Chen et al., 2020a) and closely matching the performance of M-DETR on 2D benchmarks (Kazemzadeh et al., 2014; Plummer et al., 2015) while converging twice as fast, showing its generality for language grounding across input modalities. Our code is publicly available as part of our supplemental material.

2 Related work

Object detection and referential utterance grounding. Object detection is a classic computer vision task and multiple datasets have been developed to train and evaluate detectors (Everingham et al., 2015; Lin et al., 2014a; ImageNet, 2018). A closed set of object category labels is considered and the detection model is tasked to localize *all* instances of the these object categories. While earlier architectures rely on box proposal and classification heads over convolutional variants of image encoders (He et al., 2017; Liu et al., 2015; Redmon et al., 2016), DETR (Carion et al., 2020) uses transformer architectures where a set of object query vectors attend to the scene and to one another and eventually decode objects. The recent model of d(eformable)-DETR (Zhu et al., 2021) proposes to use deformable attention, a locally adaptive kernel that is predicted directly in each pixel location without attention to other pixel locations, thus saving the quadratic cost of pixel-to-pixel attention, by noting that content-based attention does not contribute significantly in performance (Zhu et al., 2019). The works of Liu et al. (2021) and Misra et al. (2021) extend transformer encoders and detector heads to 3D point cloud input.

Even the largest object category vocabulary, such as 9000 object categories employed by the detector of Redmon & Farhadi (2017), cannot exhaustively capture visual entities in a scene. This is because visual entities appear in different levels of spatial granularity; the computer screen, the stand of the computer screen, the button on the stand of the computer screen. Most of the visual entities are ignored, but humans attend to them when required by the task (Navalpakkam & Itti, 2005).

Referential object grounding (Kazemzadeh et al., 2014), the task of localizing the object(s) referenced in a language utterance, was introduced to handle the limitation of generic object detectors to reference visual entities relevant for a task yet absent from a general vocabulary. In close inspection, object annotations of a particular category can be treated as language grounding annotations where the referential utterance is a single word, namely, the category label itself, and this is precisely exploited by our model for co-training. Early approaches (Lu et al., 2016; Xu et al., 2015; Yang et al., 2015) fuse information between utterance and pixel streams by creating a convolutional grid representation of the image then use the language to directly attend on the feature map. Later works of (Rohrbach et al., 2016; Yu et al., 2016; Fukui et al., 2016) assume given or learn to extract object proposals and then classify these proposals as being referenced by the utterance or not. This has

lately evolved into a pure bottom-up strategy of first extracting hundreds of salient regions using a pretrained object detector (Anderson et al., 2018), then feeding this set of proposals into a model that selects one of them using cross-attention with language (Lu et al., 2019; Chen et al., 2020b; Gan et al., 2020; Li et al., 2020).

Very recent works of M-DETR (Kamath et al., 2021) and Li & Sigal (2021) bypass object proposal bottlenecks using transformer encoders over image patch and word token sequences and observe large improvements in performance over previous object-bottlenecked models. This performance comes at the cost of large computation overhead due to the quadratic cost of self-attention over the number of tokens. Our model uses tripartite attention with deformable attention on the visual stream, as proposed in Zhu et al. (2021), to handle the computation overhead. Beyond attention on object proposals, it also attends to the image directly.

3D Language Grounding has only recently gained popularity (Chen et al., 2020a; Achlioptas et al., 2020). Approaches in this category resemble their 2D counterparts, but use encoders suitable for point cloud input, such as PointNet++ (Qi et al., 2017). All approaches extract object proposals, represent them as point features (Yang et al., 2021), segmentation masks (Yuan et al., 2021) or pure spatial/categorical features (Roh et al., 2021), then encode the language using embeddings (Yang et al., 2021; Roh et al., 2021) and/or scene graphs (Feng et al., 2021), to finally fuse the two representations and score each proposal using graph networks (Huang et al., 2021) or Transformers (Yang et al., 2021). Due to the difficulty of detecting objects in 3D point clouds, popular benchmarks (Achlioptas et al., 2020) evaluate using ground-truth object boxes. Our model is the first to evaluate using detected object boxes as opposed to oracle ones. Co-training of our model under both, object detection and referential grounding objectives, gives a big boost over training for grounding alone.

3 BEAUTY-DETR

Our model builds upon attention and deformable attention architectures for object detection. We describe its architecture in Section 3.1 and its training objectives in Section 3.2. We focus on grounding referential utterances in 2D and 3D, in which case the output of our model is 2D and 3D localization, respectively, for the object referents.

3.1 ARCHITECTURE

The architecture of BEAUTY-DETR is depicted in Figure 2. The model encodes a language utterance, an image or 3D point cloud, as well as a set of detected object box proposals, into separate sequences of tokens, and uses cross-attention layers to fuse information across them. After encoding, high scoring visual tokens are decoded to object boxes and are aligned to the corresponding word tokens in the utterance. In this way, all objects mentioned in the utterance are grounded to boxes in the visual scene. Please check Supplementary (Section A.1) for implementation details.

Within-modality encoding: BEAUTY-DETR can ground language in both 2D and 3D visual input. In 2D, we encode an RGB image using a pre-trained ResNet50 backbone (He et al., 2016). The 2D visual features are added with 2D Fourier positional encodings, same as in (Zhu et al., 2021; Jaegle et al., 2021). These are standard sinusoidal embeddings, as introduced in Vaswani et al. (2017), but computed in the x and y dimension separately and then concatenated. In 3D, we encode a 3D point cloud using a PointNet++ backbone (Qi et al., 2017). The 3D visual features are added with a learnable 3D positional encoding, same as Liu et al. (2021): we pass the coordinates of the points through a small MLP. In both cases, the resulting visual features are flattened to form a sequence of visual tokens, $\mathcal{V} \in \mathbb{R}^{n_v \times c_v}$, where n_v is the number of visual tokens and c_v is the number of visual feature channels.

The input visual scene is fed to a general purpose detector to obtain a set of object proposals. Following prior literature we use Faster-RCNN (Ren et al., 2015) for RGB images, pre-trained on a vocabulary of 1601 object categories on Visual Genome (Krishna et al., 2016), and Group-Free detector (Liu et al., 2021) for 3D point clouds pre-trained on a vocabulary of 485 object categories in ScanNet (Dai et al., 2017). The detected 2D and 3D box proposals that surpass a detection threshold (0.50 in 2D and 0.25 in 3D) are featurized by mapping their spatial coordinates and categorical

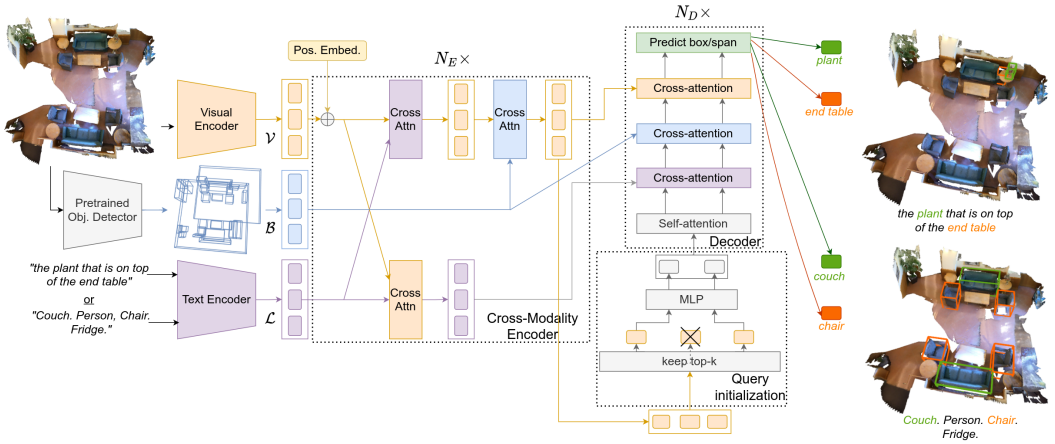


Figure 2: **BEAUTY-DETR architecture.** The input to our model is a 2D or 3D scene and a language utterance; our goal is to localize in the scene the objects that are mentioned in the utterance. The visual scene and the utterance are encoded into a sequence of tokens each using a ResNet50 (or PointNet++ in case of 3D) and RoBERTa pre-trained visual and language encoders, respectively. A pre-trained object detector extracts object box proposals that are featurized using their spatial and categorical information. At each encoder layer, visual and language tokens cross-attend and then the visual tokens attend to the detected boxes. At the end of the encoder, visual tokens are mapped to confidence scores and high-scoring tokens instantiate query vectors that will decode relevant objects. The query vectors self-attend and then attend to the encoded language tokens, detected boxes and the visual tokens. Each query eventually predicts a bounding box for an object and a span in the language utterance that the box refers to.

class information to an embedding vector each, and concatenated to form an object token. Let $\mathcal{O} \in \mathbb{R}^{n_o \times c_o}$ denote the object token sequence.

The words of the input utterance are encoded using a pre-trained RoBERTa (Liu et al., 2019) backbone, a carefully optimized version of BERT (Devlin et al., 2019) pre-trained for masked token prediction. This maps the utterance to a sequence of word tokens $\mathcal{L} \in \mathbf{R}^{n_\ell \times c_\ell}$.

All visual, word and object tokens are mapped using (different per modality) multilayer perceptrons (MLPs) to the same length feature vectors.

Cross-modality Encoder: The three modalities interact through a sequence of N_E multi-modality encoding layers comprised of self- and cross-attention operations (Lu et al., 2019). In each encoding layer, visual and language tokens cross-attend to one another and are updated using standard key-value attention. Then, the resulting language-conditioned visual tokens attend to the object tokens. In 2D images, we found it beneficial to have self-attention layers in the language and image streams using attention and deformable attention, respectively. These self-attention operations did not help in the 3D domain where the encoding layers only include cross-attention updates. We hypothesize this is due to the much smaller number of training examples available in the 3D language grounding datasets, in comparison to 2D one.

Decoder The contextualized visual tokens from the last multi-modality encoding layer are used to predict confidence scores, one per token. The top-K highest scoring tokens are each fed into an MLP to predict a vector which stands for an object query that will decode a box center and size relative to the location of the corresponding visual token. Positional encodings of the predicted box are used as positional embeddings of object query vectors. The object query vectors are updated in a residual manner through N_D decoder layers. In each decoder layer, we employ four types of attention operations. First, the object queries self-attend to one another to contextually refine their estimates. Second, the queries attend to the contextualized word embeddings to condition on the utterance. Next, the queries attend to the object proposals and then in the image or point cloud features. This way, the queries are guided by language, can select or discard the existing box proposals, and then can condition on these high-objectness areas to explore the scene as needed. At

the end of each decoding layer, there is a prediction head that predicts a box center displacement, height and width vector, and a token span for each object query that localizes the corresponding object box and aligns it with the language input. The positional embeddings of this predicted box is used as query positional embeddings for the next decoder layers, while the object query itself is just residually updated.

3.2 SUPERVISION

We supervise the outputs of all prediction heads in each layer of the decoder. Following DETR (Carion et al., 2020), we use Hungarian matching to assign a subset of queries to the ground-truth objects based on intersection-over-union between predicted and ground-truth boxes. For the queries that are matched to a ground-truth box, we use the L1 regression loss and generalized IoU (gIoU) loss (Rezatofighi et al., 2019) for the bounding box predictions.

We align detected object boxes to spans in the input utterance in a similar way to M-DETR (Kamath et al., 2021) using two objectives: i) soft token prediction for each object query that corresponds to a softmax over 256 word positions, each one corresponding to a token in the input utterance, where each query is supervised to predict a uniform distribution over all token positions that correspond to the object it is matched with and ii) contrastive matching between query embedding and word embedding vectors that ensures that the inner product of the ground-truth word-box pair embeddings scores higher than the inner product of non-corresponding word-box pairs. The query vectors that are not matched upon Hungarian matching with any ground-truth object box are set to predict "no span" and they take part in the contrastive losses as negatives. We ask the model to decode not only the "target" referent object, but also all other object mentions in the utterance, when such annotations are available. This provides denser supervision than supervising the target referent alone.

Co-training with object detection annotations Though language grounding models have effectively used supervision from multiple referential, caption description and question answering tasks, as is the case for example for the VilBERT model of Lu et al., object detection annotations have not been considered during such co-training. Yet, object detection is an instance of referential language grounding in which the utterance is a single word, namely, the object category label. Existing language and vision models only implicitly learn to ground single word utterances.

We cast object detection as the grounding of referential utterances comprised of a sequence of object category labels, as shown in Figure 2. Specifically, given a vocabulary of object category labels, we randomly sample a fixed number of them—some appear in the visual scene and some do not—and generate synthetic utterances by sequencing the sampled category labels, e.g., *"Dog. Cat. Person. Cake"*. Then, we treat these utterances as the ones to be grounded: the task is to localize all object instances of the category labels mentioned in the utterance if they appear in the scene. The sampling of negative labels category labels (labels for which there are no instances present) operates as negative training: when presented with a caption that erroneously mentions an object, the model is trained to avoid grounding of wrong labels.

4 EXPERIMENTS

We test BEAUTY-DETR on language grounding in 2D and 3D scenes. Our experiments aim to answer the following questions: **(i)** How does BEAUTY-DETR perform compared to the state-of-the-art in 2D and 3D grounding of referential expressions? **(ii)** How do different components of our model affect performance, for example, the inclusion of the object proposal stream and the inclusion of synthetic utterances from object detection annotations?

For language grounding in 3D scenes, we test the 3D version of BEAUTY-DETR on SR3D/NR3D (Achlioptas et al., 2020) and ScanRefer (Chen et al., 2020a) benchmarks. All three benchmarks contain pairs of 3D point clouds of indoor scenes from ScanNet (Dai et al., 2017) and corresponding language referential expressions, and the task is to localize in 3D the objects referenced in the utterance. The utterances in SR3D are shorter and synthetic, e.g. "Choose the couch that is underneath the picture", while utterances in NR3D and ScanRefer contain natural utterances that are longer and noisier, e.g. "From the set of chairs against the wall, the chair farthest from the red wall, in the group of chairs that is closer to the red wall". For fair comparison against previous methods, we

Method	SR3D		NR3D	ScanRefer
	Acc. (Det)	Acc. (GT)	Acc. (Det)	Acc. (Det)
ReferIt3DNet (Achlioptas et al., 2020)	27.7†	39.8	24.0†	26.4
ScanRefer (Chen et al., 2020a)	-	-	-	35.5
TGNN (Huang et al., 2021)	-	45.0	-	37.4
InstanceRefer (Yuan et al., 2021)	31.5‡	48.0	29.9‡	40.2
FFL-3DOG (Feng et al., 2021)	-	-	-	41.3
LanguageRefer (Roh et al., 2021)	39.5†	56.0	28.6†	-
TransRefer3D (He et al., 2021)	-	57.4	-	-
SAT-2D (Yang et al., 2021)*	35.4†	57.9	31.7†	44.5
BEAUTY-DETR (ours)	**48.5**	**60.4**	**34.1**	**46.4**

Table 1: **Results on language grounding in 3D point clouds.** We evaluate top-1 accuracy using ground-truth *(GT)* or detected *(Det)* boxes under 0.25 threshold. * denotes method uses extra 2D image features. † denotes evaluation with detected boxes using the authors' code and checkpoints. ‡ denotes re-training using the authors' code.

Method	Val			Test			Training Epochs	Training GPU hours	Parameters
	R@1	R@5	R@10	R@1	R@5	R@10			
VisualBERT (Li et al., 2019)	70.4	84.5	86.3	71.3	85.0	86.5	-	-	-
M-DETR (Kamath et al., 2021)	**82.5**	**92.9**	**94.9**	**83.4**	**93.5**	**95.3**	40 + 0	5480	185M
BEAUTY-DETR (ours)	81.2	90.8	92.6	81.3	91.1	92.8	**13 + 1**	**2920**	**176M**

Table 2: **Results on language grounding in Flickr30k 2D images** using Recall@k metric and computational efficiency. All training times are computed using same V100 GPU machines. Training epochs and GPU Hours are written as $x + y$ where x = number of pre-training epochs or GPU hours and y = number of fine-tuning epochs or GPU Hours.

separately train BEAUTY-DETR on each of SR3D, NR3D and ScanRefer, extended with ScanNet object detection annotations. SR3D provides annotations for all objects mentioned in the utterance, so during training we supervise localization of all object mentioned.

For language grounding in 2D scenes, we test the 2D version of BEAUTY-DETR on Flickr30k (Plummer et al., 2015) and RefCOCO (Kazemzadeh et al., 2014). We first apply the pre-training strategy of M-DETR, extended with object detection annotations from MS-COCO dataset (Lin et al., 2014b). Since pre-training is computationally expensive due to the size of the combined datasets, we do our ablations on RefCOCO without pre-training and use the best design choices for pre-training. We pretrain on combined annotations from Flickr30k (Plummer et al., 2015), MS COCO (Lin et al., 2014a), and Visual Genome Krishna et al. (2016). After pretraining, we finetune for 1 epoch on Flickr30k and 2 epochs on RefCOCO.

4.1 Results on 3D and 2D benchmarks

3D Language Grounding We compare BEAUTY-DETR to other state-of-the-art 3D language grounding approaches in Table 1. All previous methods that have been tested in SR3D or NR3D use ground-truth 3D object boxes (without labels). By design, our model does not assume oracle boxes since it directly attends to visual features and detected objects to find more objects. We thus re-train all previous models using their publicly available code, providing the same 3D object proposals we use in BEAUTY-DETR , obtained by Group-Free object detector trained to detect 485 categories in ScanNet (Section *Det* in Table-1). Additionally, we also evaluate with ground truth boxes (denoted section *Det* in Table 1) on SR3D to compare against prior work directly. The metric used is top-1 accuracy, which measures the percentage of times we can find the target box with an IoU higher than 0.25. Under all different protocols, BEAUTY-DETR outperforms existing approaches by a large margin, including the recent SAT-2D (Yang et al., 2021) that uses additional 2D image features during training. The margins are larger on the *Det* setup, since competing models fail when the referenced object is not detected. In NR3D and ScanRefer the gains for our model are smaller in comparison to SR3D since language is very complex and the language hints are harder to interpret to improve localization of object referents. We show qualitative results in Figure 3. As we show, BEAUTY-DETR both refines existing object proposals in the proposal stream, as well as proposes object boxes when they are not there by the detector. Failures of our model are included in the Appendix.

Method	val	testA	testB	Training Epochs	Training GPU Hours	Parameters
UNITER_L (Chen et al., 2020b)	81.41	87.04	74.17	-	-	-
VILLA_L (Gan et al., 2020)	82.39	87.48	74.84	-	-	-
M-DETR (Kamath et al., 2021)	**86.75**	**89.58**	**81.41**	40 + 5	5480 + 90	185M
BEAUTY-DETR (ours)	85.1	88.3	80.7	**13 + 2**	**2912**	**176M**

Table 3: **Results on language grounding in 2D RefCOCO Dataset** on accuracy metric using standard val/testA/testB splits. All training times are computed using same V100 GPU machines. Training epochs and GPU Hours are written as $x + y$ where x = number of pre-training epochs or GPU hours and y = number of fine-tuning epochs or GPU Hours.

RefCOCO (2D)		SR3D (3D)	
Method	Accuracy	Method	Accuracy
BD-no-boxes	76.25	*BD-no-condition*	33.8
BD-no-det	76.95	*BD-no-det*	44.5
BEAUTY-DETR	79.4	*BD-no-class*	44.2

Table 4: **Ablation of design choices for our model on 2D RefCOCO dataset and 3D SR3D datasets**: Note that in these results our model is trained *only* on RefCOCO and not pre-trained on multiple datasets.

2D Language Grounding In 2D, we compare our BEAUTY-DETR with other state-of-the-art approaches on Flickr30k entities (Table-2) and RefCOCO (Table-3). On Flickr30k, we report top-1, top-5 and top-10 recall, following prior literature. Our 2D BEAUTY-DETR performs on par with state-of-the-art M-DETR under the same validation protocols. However, BEAUTY-DETR converges much faster, namely in 13 vs 40 pre-training epochs, dramatically reducing the cost of pre-training. We summarize these results in Table-2. On RefCOCO, we report top-1 accuracy on the standart val/testA/testB split provided by the dataset. The results in Table-3 indicate that our model trains two times faster than M-DETR while getting comparable performance. We include qualitative results for our model in the Appendix.

4.2 Ablative analysis

In this section, we compare BEAUTY-DETR with the following variants: **i)** *BD-no-boxes* , a model identical to BEAUTY-DETR but without detected box proposals as input, **ii)** *BD-box-only* , a model identical to BEAUTY-DETR trained to select one of the input detected boxes as the answer, without attending to the visual features. We use the same detector as BEAUTY-DETR . **iii)** *BD-no-det* , a model identical to BEAUTY-DETR without co-training with object detection annotations, **iv)** *BD-no-class* , identical to BEAUTY-DETR but not considering the box proposals' classes, which is what previous works usually do (Lu et al., 2019), **v)** *BD-no-condition* , a variant where the queries are not conditioned on the contextualized visual tokens but are scene independent learned vectors, as in Kamath et al. (2021); Carion et al. (2020).

BEAUTY-DETR outperforms *BD-box-only* by 6.6% in absolute accuracy, as seen in Table 5; we split the test set of SR3D in two splits depending on whether the detector successfully detects the target object (a detected box is considered successful when it has higher than 0.25 IoU with the groundtruth 3D box). To further stress on this result, we measure the pretrained detector's recall, as the percentage of times any detected box is successful. We found this to be 69.2%, which suggests that 30.8% of the times any box-bottlenecked model will for sure fail. On the contrary, as can be

	Overall		Detected		Missed		
	Acc.	Recall	Acc.	Recall	Acc.	Recall	Epochs
BD-box-only	41.9	69.2	60.5	100.0	0.0	0.0	20
BD-no-boxes	46.7	81.7	57.5	93.1	22.4	56.4	70
BEAUTY-DETR	48.5	82.5	62.9	95.8	16.1	52.8	30

Table 5: **Performance Analysis on SR3D.** Accuracy on SR3D for our model and ablative variants depending on whether the detector did (3rd column) or failed (4th column) to detect the target. We mention the number of training epochs needed for each model to converge to optimal performance in the validation set.

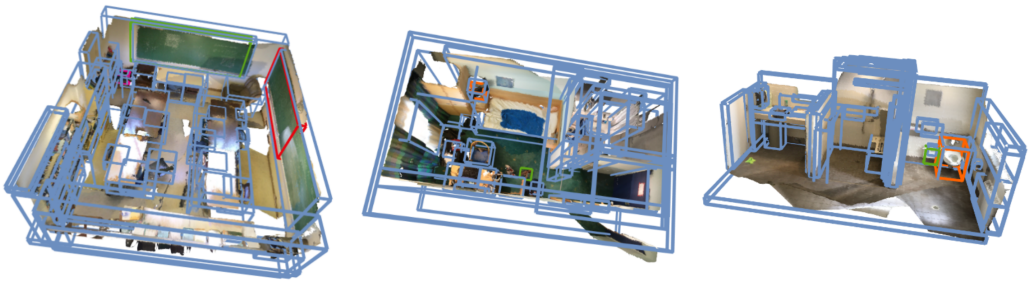


Figure 3: **Qualitative results of BEAUTY-DETR in the SR3D benchmark.** Predictions for the target are shown in green and for other mentioned objects in orange. The detected proposals appear in blue. (a) The *BD-no-boxes* variant (red box) fails to exploit the information given by the detector, but BEAUTY-DETR succeeds. (b) The detector misses the "shoes" so the *BD-box-only* variant fails. (c) The detector is successful in finding the "dustbin", still BEAUTY-DETR refines the box to get a more accurate bounding box.

seen in Table 5, BEAUTY-DETR can still work in 16.1% of the cases where an object detector fails as also shown in Figure 3b.

BEAUTY-DETR outperforms *BD-no-boxes* by 1.8% in absolute accuracy while converging in less than half the number of training epochs, 30 vs 70 (Table 5). Interestingly, the *BD-no-boxes* is slightly more robust than BEAUTY-DETR when the target box is not detected by the pre-trained detector. We hypothesize that this happens because BEAUTY-DETR learns to rely on box proposals and thus inherit its mistakes as well. However, when the object is detected, BEAUTY-DETR does better, which indicates the benefit of keeping the box proposals and thus the overall gain of BEAUTY-DETR over *BD-no-boxes* . Similar to 3D, we observe that BEAUTY-DETR is slightly better than *BD-no-boxes* and converges faster (13 epochs for *BD-no-boxes* vs 11 epochs for BEAUTY-DETR).

BEAUTY-DETR outperforms *BD-no-det* in SR3D by 4% in absolute accuracy and by 3% in RefCOCO, showing the importance of co-training with object detection annotations.

Replacing our scene-conditioned query generation with scene independent learned query vectors as in Kamath et al. (2021); Carion et al. (2020) causes a significant performance drop of 14.7% (*BD-no-condition* in Table 4).

Finally, compared to *BD-no-class* , the full BEAUTY-DETR model still has a great advantage, showing that, contrary to what detection-based approaches do, discarding class labels—that are easy and cheap to obtain from a pre-trained detector— leads to suboptimal performance.

5 Conclusion

We present BEAUTY-DETR , a model of modulated object detection in referential grounding, that attends to object proposals, language and pixel streams for localizing visual evidence to ground a language utterance. We co-train the model using both object box category annotations as well as referential utterances. The performance of our model in 2D datasets closely matches the current (and very recent) state-of-the-art while converging significantly faster, and surpasses the state-of-the-art in 3D language grounding benchmarks. BEAUTY-DETR is also the first model in 3D referential grounding that operates on the realistic setup of not having oracle object proposals available to select from, but rather detect them from the input 3D point cloud. Our extensive ablations highlight the importance of attending to bottom-up proposals without discarding the original image, and co-training with object category box annotations already available in object detection benchmarks.

6 REPRODUCIBILITY STATEMENT

We submit our code in the supplementary material to make our method reproducible. Additionally we also provide implementation details in the supplementary. We will also open-source our pre-trained checkpoints.

REFERENCES

Panos Achlioptas, Ahmed Abdelreheem, Fei Xia, Mohamed Elhoseiny, and Leonidas Guibas. ReferIt3D: Neural Listeners for Fine-Grained 3D Object Identification in Real-World Scenes. In Proc. ECCV, 2020.

Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering. In Proc. CVPR, 2018.

Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. Learning to compose neural networks for question answering. In North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL), 2016.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-End Object Detection with Transformers. In Proc. ECCV, 2020.

Dave Zhenyu Chen, Angel Chang, and Matthias Nießner. ScanRefer: 3D Object Localization in RGB-D Scans using Natural Language. In Proc. ECCV, 2020a.

Wenhu Chen, Zhe Gan, Linjie Li, Yu Cheng, William Wang, and Jingjing Liu. Meta module network for compositional visual reasoning. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 655–664, 2021.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. UNITER: UNiversal Image-TExt Representation Learning. In Proc. ECCV, 2020b.

Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas A. Funkhouser, and Matthias Nießner. ScanNet: Richly-Annotated 3D Reconstructions of Indoor Scenes. In Proc. CVPR, 2017.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, K. Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In Proc. CVPR, 2009.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proc. NAACL, 2019.

M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. International Journal of Computer Vision, 111(1):98–136, January 2015.

Hao Fang, Saurabh Gupta, Forrest Iandola, Rupesh K Srivastava, Li Deng, Piotr Dollár, Jianfeng Gao, Xiaodong He, Margaret Mitchell, John C Platt, et al. From captions to visual concepts and back. In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1473–1482, 2015.

Mingtao Feng, Zhen Li, Qi Li, Liang Zhang, XiangDong Zhang, Guangming Zhu, Hui Zhang, Yaonan Wang, and Ajmal Mian. Free-form Description Guided 3D Visual Graph Network for Object Grounding in Point Cloud. In Proc. ICCV, 2021.

Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding. In Proc. EMNLP, 2016.

Zhe Gan, Yen-Chun Chen, Linjie Li, Chen Zhu, Yu Cheng, and Jingjing Liu. Large-Scale Adversarial Training for Vision-and-Language Representation Learning. In Proc. NeurIPS, 2020.

Dailan He, Yusheng Zhao, Junyu Luo, Tianrui Hui, Shaofei Huang, Aixi Zhang, and Si Liu. TransRefer3D: Entity-and-Relation Aware Transformer for Fine-Grained 3D Visual Grounding. ArXiv, abs/2108.02388, 2021.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In Proc. CVPR, 2016.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. CoRR, abs/1703.06870, 2017. URL http://arxiv.org/abs/1703.06870.

Ronghang Hu, Marcus Rohrbach, Jacob Andreas, Trevor Darrell, and Kate Saenko. Modeling relationships in referential expressions with compositional modular networks. 11 2016.

Ronghang Hu, Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Kate Saenko. Learning to reason: End-to-end module networks for visual question answering. In IEEE International Conference on Computer Vision (ICCV), 2017.

Pin-Hao Huang, Han-Hung Lee, Hwann-Tzong Chen, and Tyng-Luh Liu. Text-Guided Graph Neural Networks for Referring 3D Instance Segmentation. In Proc. AAAI, 2021.

ImageNet. Imagenet object localization challenge, 2018. URL https://www.kaggle.com/c/imagenet-object-localization-challenge.

Andrew Jaegle, Felix Gimeno, Andrew Brock, Andrew Zisserman, Oriol Vinyals, and João Carreira. Perceiver: General Perception with Iterative Attention. In Proc. ICML, 2021.

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and Ross B. Girshick. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. CoRR, abs/1612.06890, 2016a. URL http://arxiv.org/abs/1612.06890.

Justin Johnson, Andrej Karpathy, and Li Fei-Fei. Densecap: Fully convolutional localization networks for dense captioning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016b.

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Judy Hoffman, Fei-Fei Li, C. Lawrence Zitnick, and Ross B. Girshick. Inferring and executing programs for visual reasoning. CoRR, abs/1705.03633, 2017. URL http://arxiv.org/abs/1705.03633.

Aishwarya Kamath, Mannat Singh, Yann André LeCun, Ishan Misra, Gabriel Synnaeve, and Nicolas Carion. MDETR - Modulated Detection for End-to-End Multi-Modal Understanding. In Proc. ICCV, 2021.

Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. IEEE Trans. Pattern Anal. Mach. Intell., 39(4):664–676, April 2017. ISSN 0162-8828. doi: 10.1109/TPAMI.2016.2598339. URL https://doi.org/10.1109/TPAMI.2016.2598339.

Fumi Katsuki and Christos Constantinidis. Bottom-up and top-down attention: different processes and overlapping neural systems. The Neuroscientist, 20(5):509–521, 2014.

Sahar Kazemzadeh, Vicente Ordonez, Marc André Matten, and Tamara L. Berg. ReferItGame: Referring to Objects in Photographs of Natural Scenes. In Proc. EMNLP, 2014.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. International Journal of Computer Vision, 123, 2016.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. VisualBERT: A Simple and Performant Baseline for Vision and Language. ArXiv, abs/1908.03557, 2019.

Muchen Li and Leonid Sigal. Referring Transformer: A One-step Approach to Multi-task Visual Grounding. ArXiv, abs/2106.03089, 2021.

Xiujun Li, Xi Yin, Chunyuan Li, Xiaowei Hu, Pengchuan Zhang, Lei Zhang, Lijuan Wang, Houdong Hu, Li Dong, Furu Wei, Yejin Choi, and Jianfeng Gao. Oscar: Object-Semantics Aligned Pre-training for Vision-Language Tasks. In Proc. ECCV, 2020.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. CoRR, abs/1405.0312, 2014a. URL http://arxiv.org/abs/1405.0312.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common Objects in Context. In Proc. ECCV, 2014b.

Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal Loss for Dense Object Detection. In Proc. ICCV, 2017.

Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: single shot multibox detector. CoRR, abs/1512.02325, 2015. URL http://arxiv.org/abs/1512.02325.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, M. Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. ArXiv, abs/1907.11692, 2019.

Ze Liu, Zheng Zhang, Yue Cao, Han Hu, and Xin Tong. Group-Free 3D Object Detection via Transformers. ArXiv, abs/2104.00678, 2021.

Jiasen Lu, Vedanuj Goswami, Marcus Rohrbach, Devi Parikh, and Stefan Lee. 12-in-1: Multi-Task Vision and Language Representation Learning. In 2020.

Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical Question-Image Co-Attention for Visual Question Answering. In Proc. NIPS, 2016.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks. In Proc. NeurIPS, 2019.

Jiayuan Mao, Chuang Gan, Pushmeet Kohli, Joshua B. Tenenbaum, and Jiajun Wu. The Neuro-Symbolic Concept Learner: Interpreting Scenes, Words, and Sentences From Natural Supervision. In International Conference on Learning Representations, 2019. URL https://openreview.net/forum?id=rJgMlhRctm.

Ishan Misra, Rohit Girdhar, and Armand Joulin. An End-to-End Transformer Model for 3D Object Detection. In Proc. ICCV, 2021.

Vidhya Navalpakkam and Laurent Itti. Modeling the influence of task on attention. Vision Research, 45(2):205–231, 2005. ISSN 0042-6989. doi: https://doi.org/10.1016/j.visres.2004.07.042. URL https://www.sciencedirect.com/science/article/pii/S004269890400392X.

Bryan A. Plummer, Liwei Wang, Christopher M. Cervantes, Juan C. Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k Entities: Collecting Region-to-Phrase Correspondences for Richer Image-to-Sentence Models. In Proc. ICCV, 2015.

Charles Qi, Li Yi, Hao Su, and Leonidas J. Guibas. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In Proc. NIPS, 2017.

Joseph Redmon and Ali Farhadi. Yolo9000: Better, faster, stronger. In 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6517–6525, 2017. doi: 10.1109/CVPR.2017.690.

Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779–788, 2016.

Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. CoRR, abs/1506.01497, 2015. URL http://arxiv.org/abs/1506.01497.

Seyed Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian D. Reid, and Silvio Savarese. Generalized Intersection Over Union: A Metric and a Loss for Bounding Box Regression. In Proc. CVPR, 2019.

Junha Roh, Karthik Desingh, Ali Farhadi, and Dieter Fox. LanguageRefer: Spatial-Language Model for 3D Visual Grounding. ArXiv, abs/2107.03438, 2021.

Anna Rohrbach, Marcus Rohrbach, Ronghang Hu, Trevor Darrell, and Bernt Schiele. Grounding of Textual Phrases in Images by Reconstruction. In Proc. ECCV, 2016.

Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In Proc. NIPS, 2017.

Ke Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C. Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. In Proc. ICML, 2015.

Zhengyuan Yang, Songyang Zhang, Liwei Wang, and Jiebo Luo. SAT: 2D Semantics Assisted Training for 3D Visual Grounding. In Proc. ICCV, 2021.

Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alexander J. Smola. Stacked attention networks for image question answering. CoRR, abs/1511.02274, 2015. URL http://arxiv.org/abs/1511.02274.

Kexin Yi, Jiajun Wu, Chuang Gan, Antonio Torralba, Pushmeet Kohli, and Joshua B. Tenenbaum. Neural-symbolic vqa: Disentangling reasoning from vision and language understanding. In NeurIPS, 2018.

Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B. Tenenbaum. CLEVRER: collision events for video representation and reasoning. CoRR, abs/1910.01442, 2019. URL http://arxiv.org/abs/1910.01442.

Licheng Yu, Patrick Poirson, Shan Yang, Alexander C. Berg, and Tamara L. Berg. Modeling Context in Referring Expressions. In Proc. ECCV, 2016.

Zhihao Yuan, Xu Yan, Yinghong Liao, Ruimao Zhang, Zhen Li, and Shuguang Cui. InstanceRefer: Cooperative Holistic Understanding for Visual Grounding on Point Clouds through Instance Multi-level Contextual Referring. In Proc. ICCV, 2021.

Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Revisiting visual representations in vision-language models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5579–5588, June 2021.

Xizhou Zhu, Dazhi Cheng, Zheng Zhang, Stephen Ching-Feng Lin, and Jifeng Dai. An Empirical Study of Spatial Attention Mechanisms in Deep Networks. In Proc. ICCV, 2019.

Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: Deformable Transformers for End-to-End Object Detection. In Proc. ICLR, 2021.

A APPENDIX

A.1 IMPLEMENTATION DETAILS

We report here architecture choices as well as training hyperparameters. We implement BEAUTY-DETR in PyTorch. For the 2D version, the image is encoded using ResNet-50 (He et al., 2016) pretrained on ImageNet (Deng et al., 2009). We use multi-scale features as in Zhu et al. (2021). The feature maps of the different scales are flattened and concatenated in the spatial dimension, leading to 17821 visual tokens. The feature dimension of each token is 256. To obtain the box proposals, we use the detector of Anderson et al. (2018) trained on 1601 classes of Visual Genome (Krishna et al., 2016). The detected boxes are encoded using their spatial and categorical features. Specifically, we compute the 2D Fourier features of each box and feed them to an MLP, then we concatenate this vector with a learnable semantic class embedding and feed to another MLP to obtain the box embeddings. To form queries, we rank visual tokens based on their confidence score and keep the 300 most confidence ones. This confidence layer is supervised using Focal Loss (Lin et al., 2017): we assign a positive objectness scores to every point that lies inside a ground-truth answer box. We set $N_E = 6$ and $N_D = 6$. All attention layers to the visual stream are implemented with deformable attention (Zhu et al., 2021), attention to other the language stream or detected boxes is the standard attention of Vaswani et al. (2017) and Lu et al. (2019).

For the 3D version, the point cloud is encoded with PointNet++ (Qi et al., 2017) using the same hyperparameters as in (Liu et al., 2021), pre-trained on ScanNet (Dai et al., 2017). We use the last layer's features, resulting in 1024 visual tokens. In the cross-modality encoder, instead of allowing the visual features to attend to the box features, we directly concatenated the box features to the input point cloud. Specifically, for all the points that lie inside a box, we concatenate this box's features directly to their point features (xyz and color). If a point lies inside multiple boxes, we randomly sample one box's features. Points that do not lie inside inside any box are padded with zeros. This is computationally cheaper than cross-attending visual features to box features and works well in 3D since the objects do not intersect. In 2D, however, it does not work well since the objects and thus their boxes overlap a lot and hence usually a pixel falls inside multiple boxes. In decoder, the queries are formed from the top 256 most confident visual tokens similar to the 2D version. We set $N_E = 3$ with no self-attention layers, $N_D = 6$. All attention layers are implemented using standard self-/cross-attention.

For the 2D model we use a learning rate of $1e{-}6$ for Resnet50 visual encoder, $5e{-}6$ for RoBERTa text encoder and $1e{-}5$ for rest of the layers. We pre-train on 4 V100s with a batch size of 1, and finetune on RefCOCO with a batch size of 3 on 4 V100s. For the 3D, we use a learning rate of $1e{-}5$ for RoBERTa and $1e{-}4$ for all other layers. We are able to fit a batch size of 6 on a single GPU of 12GB. We will release pre-trained checkpoints for both 2D and 3D.

A.2 QUALITATIVE RESULTS

We show qualitative results of the 2D version of BEAUTY-DETR on RefCOCO in Figure 4. We also show failure cases on SR3D in Figure 5.

(a) woman in the right of the stands with her hands crossed

(b) man on left in orange

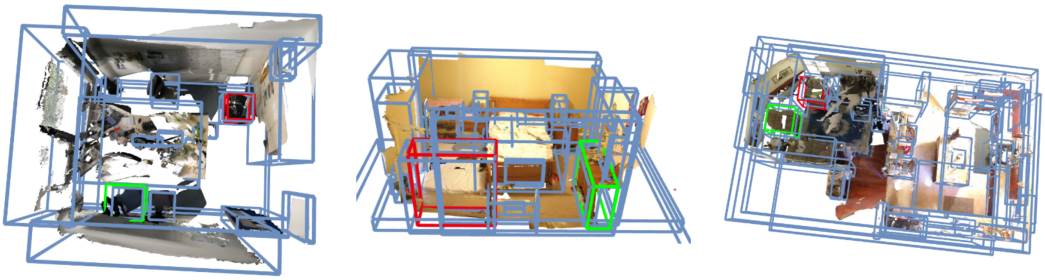
(c) little kid in the snow and in front

(d) right cow with white fur

(e) white chair top right

(f) orange and black boat on green water

Figure 4: Qualitative results of BEAUTY-DETR on RefCOCO. Our model is robust even in cases the detector has not captured the correct answer.



(a) the office chair that is beside the chair

(b) find the dresser that is next to the trash can

(c) find the armchair that is next to the table

Figure 5: Failure cases of BEAUTY-DETR on SR3D. Our predictions with red, ground-truth with green. Even if the box is there, still our model can fail, proving that ranking the correct boxes over other proposals remains a hard problem.