See It All: Contextualized Late Aggregation for 3D Dense Captioning

Minjung Kim[1,*] **Hyung Suk Lim**[1,3] **Seung Hwan Kim**[2] **Soonyoung Lee**[2]
Bumsoo Kim[2,†] **Gunhee Kim**[1,†]

[1]Seoul National University [2]LG AI Research [3]Diquest

[1]minjung.kim@vision.snu.ac.kr, [3]hslim@diquest.com, [2]bumsoo.kim@lgresearch.ai, [1]gunhee@snu.ac.kr

Abstract

3D dense captioning is a task to localize objects in a 3D scene and generate descriptive sentences for each object. Recent approaches in 3D dense captioning have adopted transformer encoder-decoder frameworks from object detection to build an end-to-end pipeline without hand-crafted components. However, these approaches struggle with contradicting objectives where a single query attention has to simultaneously view both the tightly localized object regions and contextual environment. To overcome this challenge, we introduce SIA (See-It-All), a transformer pipeline that engages in 3D dense captioning with a novel paradigm called late aggregation. SIA simultaneously decodes two sets of queries—context query and instance query. The instance query focuses on localization and object attribute descriptions, while the context query versatilely captures the region-of-interest of relationships between multiple objects or with the global scene, then aggregated afterwards (i.e., late aggregation) via simple distance-based measures. To further enhance the quality of contextualized caption generation, we design a novel aggregator to generate a fully informed caption based on the surrounding context, the global environment, and object instances. Extensive experiments on two of the most widely-used 3D dense captioning datasets demonstrate that our proposed method achieves a significant improvement over prior methods.

1 Introduction

3D dense captioning has been defined in former works (Chen et al., 2021, 2022; Yuan et al., 2022; Wang et al., 2022; Jiao et al., 2022; Cai et al., 2022; Chen et al., 2023a) as the task of localizing all the objects in a 3D scene (i.e., object detection) and generating descriptive sentences for each object (i.e., object caption generation). Early works incorporated a two-stage "detect-then-describe" pipeline, where we first detect all the object proposals then generate the captions for each object (Chen et al., 2021; Jiao et al., 2022; Wang et al., 2022; Zhong et al., 2022; Cai et al., 2022; Chen et al., 2022; Yuan et al., 2022). However, the sequential design, lacking sufficient integration of contextual information in these endeavors, has been limited in performance and efficiency.

Vote2Cap-DETR (Chen et al., 2023a) has emulated the transformer encoder-decoder pipeline from object detection (Carion et al., 2020) to alleviate these issues and fashioned an end-to-end pipeline for 3D dense captioning. Powered by transformer attentions, this method contextualizes individual objects (i.e., self-attention with other proposals throughout the global scene) to generate dense captions. Nevertheless, compared to the notable advancements that object detection has experienced, the direct application of this architecture has failed to fully leverage the contextual information required for 3D dense captioning.

Dense captioning has to perform precise object localization while generating captions that either independently describe an object's attributes (e.g., a *wooden* chair) or describe the object within its contextual environment (e.g., a chair *in front of* the TV). This presents a challenging scenario where the feature representation for a single query must encompass both accurate local features for localization or attribute-based caption generation, alongside incorporating contextual features that dynamically span neighboring regions or the broader global scene. Focusing attention on local features can enhance localization and detailed attribute description but reduce sensitivity to the surrounding context. Conversely, spreading attention to include the context can improve understanding of the environmental description but at the cost of localization accuracy.

In this paper, we propose a pipeline engaging a novel *late aggregation* paradigm called **SIA (i.e.,**

[*] Work done during internship at LG AI Research
[†] Corresponding authors

Findings of the Association for Computational Linguistics ACL 2024, pages 3395–3405
August 11-16, 2024

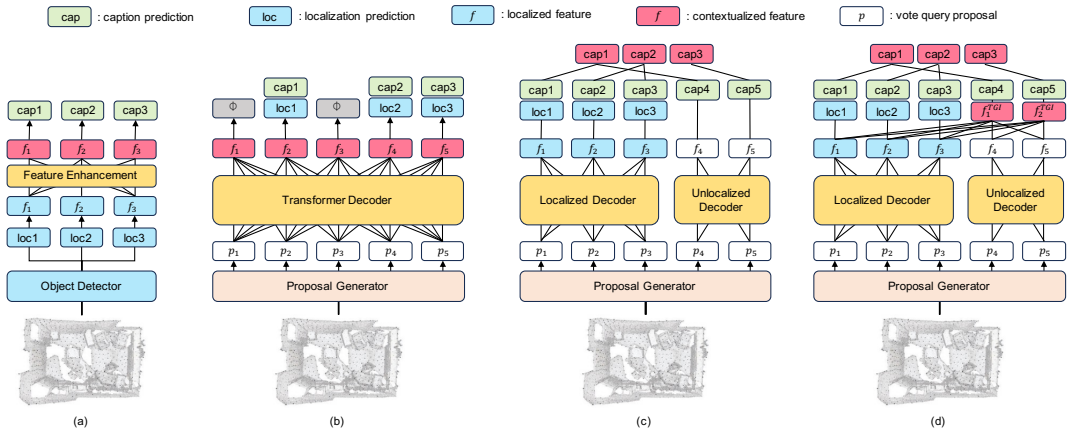

Figure 1: Schematic diagrams illustrating paradigms of 3D dense captioning: (a) features are extracted from object detectors, and their relations are further aggregated to enhance features Cai et al. (2022) (b) proposals are generated by voting, then the local-context features are aggregated with transformer attention Chen et al. (2023a) (c) our proposed SIA separately encodes features with local boundaries and context features without such boundaries, and aggregates the generated caption that involves identical objects afterward (i.e., *late aggregation*) (d) SIA with further enhanced contextual features generated from our novel TGI-Aggregator (f^{TGI}) that aggregates local-context-global features for a more contextualized caption generation.

See It All). Rather than assigning a query *per object* and training them to dynamically incorporate local features, contextual information, and the global scene, SIA allocates a query *per caption*. To elaborate, SIA identifies a distinct region to focus on when generating each caption and then consolidates the outcomes concerning identical objects. Figure 1 contrasts our proposed late aggregation approach with existing dense captioning paradigms. While previous works have either (a) extracted the features from localized object areas Cai et al. (2022) or (b) generated captions from features that have to perform both localization and proper caption prediction Chen et al. (2023a), SIA (c) focuses on each unique ROIs for each caption then aggregates the captions that include identical objects afterwards. This architecture (i.e., *late aggregation*) enables SIA to produce attribute captions and localization with features concentrated exclusively on specific local areas, while captions necessitating a broad range of contextual information can be crafted using features gathered without the constraints of localization boundaries.

To further refine the features for contextual captions, we design a unique aggregator that generates captions based on the con**T**extual surroundings, **G**lobal descriptor, and **I**nstance features (i.e., TGI-Aggregator, see Figure 1-(d)). Our TGI-Aggregator generates contextual caption based on the fully informed feature that can dynamically capture the area of interest within the scene without the constraints of localization objectives. Extensive experiments on two widely used benchmarks in 3D dense captioning (i.e., ScanRefer (Chen et al., 2020a) and Nr3D (Achlioptas et al., 2020)) show that our proposed SIA surpasses prior approaches by a large margin. The contribution of our paper can be summarized as:

- We propose a new paradigm for 3D dense captioning (i.e., *late aggregation*). While previous works aggregate the instance and context features first and then generate captions, SIA generates local and contextual captions separately and then aggregates the captions involving identical objects.
- To further improve the quality of features used for contextualized caption generation, we propose a novel aggregator named TGI-Aggregator.
- Our SIA achieves state-of-the-art performances across multiple evaluation metrics on the ScanRefer and Nr3D datasets.

2 Preliminary

In this preliminary, we start with a basic transformer-based end-to-end 3D dense captioning pipeline (Chen et al., 2023a). The caption head is attached to the top of the existing 3D object detection pipeline (Qi et al., 2019; Misra et al., 2021)

with vote queries that establish captions for each pinpointed object throughout the scene in an object-by-object manner. Afterward, we discuss why this object-centric application of transformer attention is unsuitable for 3D dense captioning.

2.1 End-to-End 3D Object Detection

3D object detection aims to identify and localize objects in 3D scenes. VoteNet (Qi et al., 2019) incorporates an encoder-decoder architecture where the bounding boxes are predicted by aggregating the *votes* for the center coordinates. 3DETR (Misra et al., 2021) generates *object queries* by uniformly sampling seed points from a 3D scene. Vote2Cap-DETR (Chen et al., 2023a) uses *vote queries* that connect the object queries in 3DETR to VoteNet, resulting in better localization and improved training efficiencies.

2.2 Extension to 3D Dense Captioning

The goal of 3D dense captioning is to localize objects in a 3D scene and generate informative natural language descriptions per object. An intuitive extension from object detection to 3D dense captioning is simply applying a captioning head for each object proposals (Chen et al., 2023a). Given an input indoor 3D scene as a point cloud $PC = [p_{\text{in}}; f_{\text{in}}] \in \mathbb{R}^{N\times(3+F)}$, where $p_{\text{in}} \in \mathbb{R}^{N\times 3}$ is the absolute locations for each point and $f_{\text{in}} \in \mathbb{R}^{N\times F}$ is additional input features for each point (Chen et al., 2020a, 2021), the objective of 3D dense captioning is to generate a set of box-caption pairs $(\hat{B}, \hat{C}) = \{(\hat{b_1}, \hat{c_1}), ..., (\hat{b_K}, \hat{c_K})\}$, representing an estimation of K distinctive objects in this 3D scene. Captions are generated in *parallel* with bounding box prediction using a caption head. Since the aforementioned vote queries (i.e., p_{vote}) fail to provide adequate attributes and spatial relations for informative caption generation, the contextual information is leveraged through a separate lightweight transformer (Chen et al., 2023a).

2.3 Retrospect on Object-Centric Captioning

Current 3D dense captioning benchmarks require the model to generate multiple captions for each detected object. Therefore, it seems natural to approach this task in an *object-centric* manner (Wang et al., 2022; Jiao et al., 2022; Achlioptas et al., 2020), where we generate captions per each object proposal. However, unlike object detection, dense captioning requires an extensive understanding of the scene, including the attributes of each object and the relative information between objects and the global scene. Therefore, designating the queries per object requires a single query attention to versatilely encompass the individual object and its surrounding elements, failing to concentrate on the local element it should describe effectively. We propose a novel *late aggregation* approach for 3D dense captioning to address this issue and incorporate contextual scene information.

3 Method

In this section, we introduce a transformer encoder-decoder pipeline that engages our novel *late aggregation* paradigm for 3D dense captioning. In previous methods, the transformer attention aggregates contextual information *per-object*, where a single feature is used to perform localization, generate localized attributes and simultaneously capture the surrounding context area. SIA is designed to capture the unique region of interest for each caption. Local attribute descriptions are generated with localized features. In contrast, contextualized captions that include relationships with other objects or the entire scene are generated with a separately decoded feature irrelevant to localization objectives. Then, captions involving identical objects are aggregated via distance (i.e., *late aggregation*) to consist of the final caption. The overall pipeline is illustrated in Figure 2.

3.1 Encoder

Given the input point cloud $PC = [p_{\text{in}}; f_{\text{in}}] \in \mathbb{R}^{N\times(3+F)}$, the input point cloud is first tokenized by a set-abstraction layer of PointNet++ (Qi et al., 2017). The tokenized output is inputted into a masked transformer encoder with the set-abstraction layer, followed by two additional encoder layers. The final encoded scene tokens are denoted as $p_{enc} \in \mathbb{R}^{1,024\times 3}$ and $f_{enc} \in \mathbb{R}^{1,024\times 256}$.

3.2 Context Query and Instance Query

To disentangle the captions that are bound to a single object and captions that include relative information with other objects or the global scene, we designate two separate *instance query* and *context query* to each capture a unique region per caption within the 3D scene. While the context query captures the local-global regions capable of captioning, the *instance query* generates standard object localization and attribute-related caption prediction for each object. The two queries are decoded in parallel and later aggregated to consist of the final caption.

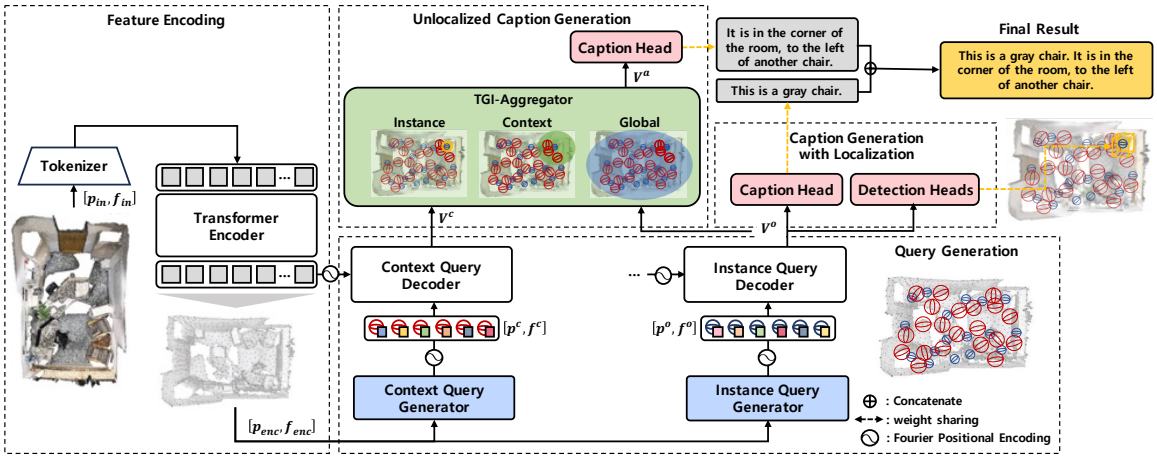

Figure 2: Overall architecture of SIA for 3D dense captioning. The caption query set is each designated to Instance Query Decoder and Context Query Decoder. In the Instance Query Decoder, the caption based on the tight localized area are generated along with object detection. In the Context Query Decoder, captions that require views transcending single object localization such as captions containing relation between multiple objects or relation between the scene are generated. The feature for this Unlocalized Caption Generation is further enhanced with our novel TGI-Aggregator, that contextualizes the feature from con**T**ext regions, the **G**lobal scene, and **I**nstances.

Context Query Generator. Given the encoded scene tokens $(p_{\text{enc}}, f_{\text{enc}})$, we sample 512 context points p^c_{seed} with farthest point sampling (FPS) on p_{enc}. Then, the context query (p^c, f^c) is represented as:

$$(p^c, f^c) = \text{SA}_c(p_{\text{enc}}, f_{\text{enc}}), \quad (1)$$

where SA_c denotes the set-abstraction layer (Qi et al., 2017) with a radius of 1.2 and samples 64 points for p^c.

Instance Query Generator. The instance query is decoded to perform standard 3D object detection and generate captions for the individual attributes of each object. Likewise, the instance query (p^o, f^o) is written as:

$$[\Delta p_{\text{vote}}; \Delta f_{\text{vote}}] = \text{FFN}_o(f_{\text{enc}}), \quad (2)$$

$$(p^o, f^o) = \text{SA}_o\big(p_{\text{enc}} + \Delta p_{\text{vote}}, f_{\text{enc}} + \Delta f_{\text{vote}}\big), \quad (3)$$

where $[\Delta p_{\text{vote}}; \Delta f_{\text{vote}}] \in \mathbb{R}^{1,024\times(3+256)}$ is an offset that learns to shift the encoded points to object's centers spatially by a feed-forward network FFN_o, following (Chen et al., 2023a). SA_o denotes the set-abstraction layer with a radius of 0.3 and samples 16 points for p^o. All hyper-parameters are set experimentally.

3.3 Decoding

Given the context and instance queries, we build a parallel decoding pipeline where the Context Decoder describes contextual information between objects, and the Instance Decoder performs the localization and attribute description. We then feed the decoded context query V^c and instance query V^o to our TGI-Aggregator.

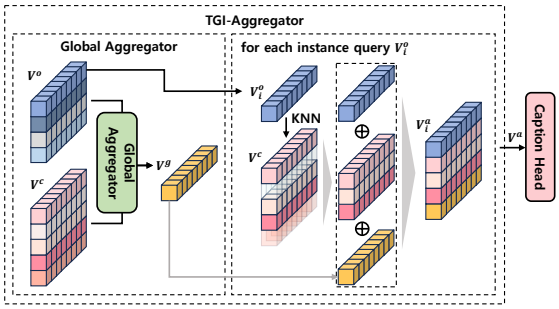

Figure 3: Conceptual illustration of our TGI-Aggregator. The Global Aggregator $G(\cdot)$ aggregates the decoded context query V^o and instance query V^c to construct a global descriptor V^g. Then, the instance feature V^o_i, the nearest neighbor feature in V^c, and the global descriptor V^g are concatenated to construct V^a.

TGI-Aggregator. Figure 3 shows a conceptual illustration of our TGI-Aggregator. To encompass the understanding of the entire scene for each caption, we generate a global feature using all decoded context queries V^c and instance queries V^o; related experiments can be found in Section 4.4. We deploy a clustering-based aggregator (Arandjelovic et al., 2016) $G(\cdot)$. As a result, we obtain a global descriptor $V^g \in \mathbb{R}^{256}$ by $V^g = G(V^c, V^o)$. Then, we concatenate this global descriptor V^g to each

decoded instance query V_i^o and K nearest features within V^c in terms of spatial proximity to V_i^o, resulting in an aggregated feature V^a that contains a comprehensive information of con**T**ext, **G**lobal, and **I**nstance. We set K to 16.

Contextual Caption Generation. For caption generation, we adopt a transformer decoder-based caption head based on GPT-2 (Radford et al., 2019), following Vote2Cap-DETR (Chen et al., 2023a) and SpaCap3D (Wang et al., 2022). The output V^a of our TGI-Aggregator contains features about contextual surroundings, global context, and instance information. Based on this feature, SIA can generate descriptions for relationships between multiple objects (e.g., the chair is next to a bookshelf) and global relationships (e.g., the table is in the middle of the room).

Localization & Attribute Caption Generation. To predict localization and attribute descriptions for the participating instances, we feed the decoded instance query V^o and parallelly feed it into the detection head and (shared) caption head. For localization, we follow 3DETR (Misra et al., 2021), reformulating the box corner estimation as offset estimation from a query point to an object's center and box size regression. All subtasks are implemented by FFNs. The object localization head is shared throughout the decoder layers.

3.4 Training SIA

We construct the final caption to be object-centric to compare with previous object-centric methods using benchmark datasets. The final caption for the i-th object is obtained by simply concatenating the captions generated from V_i^o and V_i^a. Our SIA is trained and evaluated by locating all objects within a scene and comparing the final caption centered on each object with the ground-truth.

Instance Query Loss. To train the Instance Query Generator to find an object's center by shifting points p_{enc}, we adopt the vote loss from VoteNet (Qi et al., 2019). Given the generated instance query (p^o, f^o) and the encoded scene tokens (p_{enc}, f_{enc}), the vote loss $\mathcal{L}^o$ is written as:

$$\mathcal{L}^o = \frac{1}{M}\sum_{i=1}^{M}\sum_{j=1}^{N_{gt}} \|p_i^o - \text{cnt}_j\|_1 \cdot \mathbb{I}(p_{enc}^i), \quad (4)$$

where $\mathbb{I}(x)$ is an indicator function that equals 1 when $x \in I_j$ and 0 otherwise, N_{gt} is the number of instances in a 3D scene, M is the number of p^o, and cnt_j is the center of j-th instance I_j.

Detection Loss. We use Hungarian matching (Kuhn, 1955) to assign each proposal with the ground-truth, following DETR (Carion et al., 2020). The detection loss $\mathcal{L}_{\text{det}}$ is written as:

$$\mathcal{L}_{\text{det}} = \alpha_1\mathcal{L}_{\text{giou}} + \alpha_2\mathcal{L}_{\text{cls}} + \alpha_3\mathcal{L}_{\text{cnt}} + \alpha_4\mathcal{L}_{\text{size}}, \quad (5)$$

where $\alpha_1 = 10, \alpha_2 = 1, \alpha_3 = 5, \alpha_4 = 1$ are set heuristically. The detection loss is applied across all decoder layers for better convergence.

Caption Loss. Following the standard protocol for image captioning, we first train caption heads with standard cross-entropy loss for Maximum Likelihood Estimation (MLE). In the MLE training, the model learns to predict the $(t+1)$-th word c_n^{t+1} based on the first t words $c_n^{[1:t]}$ and the visual features $\mathcal{V}$. The loss function is established for the final caption with length T is defined as follows:

$$\mathcal{L}_{c_n} = \sum_{t=1}^{T}\mathcal{L}_{c_n}(t) = -\sum_{t=1}^{T}\log\hat{P}\left(c_n^{t+1}|\mathcal{V}, c_n^{[1:t]}\right), \quad (6)$$

Once the caption head is trained with word-level supervision, it is refined using Self-Critical Sequence Training (SCST) (Rennie et al., 2017). In this phase, the model produces multiple captions $\hat{c}_{1,\ldots,k}$ with a beam size of k and an additional $\hat{g}$ using greedy search as a baseline. The loss function for SCST is formulated as follows:

$$\mathcal{L}_{c_n} = -\sum_{i=1}^{k}(R(\hat{c}_i) - R(\hat{g})) \cdot \frac{1}{|\hat{c}_i|}\log\hat{P}(\hat{c}_i|\mathcal{V}). \quad (7)$$

The reward function $R(\cdot)$ is based on the CIDEr (Vedantam et al., 2015) metric for caption evaluation, and the logarithmic probability of the caption $\hat{c}_i$ is normalized by caption length $|\hat{c}_i|$, promoting equal importance to captions of varying lengths by the model.

Final Loss for SIA. Given the instance query Loss $\mathcal{L}^o$, the detection loss for the i-th decoder layer as $\mathcal{L}_{\text{det}}^i$, and the average of the caption loss $\mathcal{L}_{c_n}$ within a batch denoted as $\mathcal{L}_{\text{cap}}$, the final loss $\mathcal{L}$ for SIA is written as:

$$\mathcal{L} = \beta_1\mathcal{L}^o + \beta_2\sum_{i=1}^{n_{\text{dec-layer}}}\mathcal{L}_{\text{det}}^i + \beta_3\mathcal{L}_{\text{cap}}, \quad (8)$$

where $\beta_1 = 10$, $\beta_2 = 1$, and $\beta_3 = 10$.

Model	Training	w/o additional 2D data								w/ additional 2D data							
		IoU=0.25				IoU=0.50				IoU=0.25				IoU=0.50			
		C↑	B-4↑	M↑	R↑	C↑	B-4↑	M↑	R↑	C↑	B-4↑	M↑	R↑	C↑	B-4↑	M↑	R↑
Scan2Cap	MLE	53.73	34.25	26.14	54.95	35.20	22.36	21.44	43.57	56.82	34.18	26.29	55.27	39.08	23.32	21.97	44.78
D3Net		-	-	-	-	-	-	-	-	-	-	-	-	46.07	30.29	24.35	51.67
SpaCap3d		58.06	35.30	26.16	55.03	42.76	25.38	22.84	45.66	63.30	36.46	26.71	55.71	44.02	25.26	22.33	45.36
MORE		58.89	35.41	26.36	55.41	38.98	23.01	21.65	44.33	62.91	36.25	26.75	56.33	40.94	22.93	21.66	44.42
3DJCG		60.86	39.67	27.45	59.02	47.68	31.53	24.28	51.80	64.70	40.17	27.66	59.23	49.48	31.03	24.22	50.80
Contextual		-	-	-	-	42.77	23.60	22.05	45.13	-	-	-	-	46.11	25.47	22.64	45.96
REMAN		-	-	-	-	-	-	-	-	62.01	36.37	27.76	56.25	45.00	26.31	22.67	46.96
3D-VLP		64.09	39.84	27.65	58.78	50.02	31.87	24.53	51.17	70.73	41.03	28.14	59.72	54.94	32.31	24.83	51.51
Vote2Cap-DETR		71.45	39.34	28.25	59.33	61.81	34.46	26.22	54.40	72.79	39.17	28.06	59.23	59.32	32.42	25.28	52.38
Unit3D		-	-	-	-	-	-	-	-	-	-	-	-	46.69	27.22	21.91	45.98
Ours		**78.68**	**43.25**	**29.21**	**63.06**	**73.22**	**40.91**	**28.19**	**60.46**	**78.05**	**42.16**	**28.74**	**61.70**	**69.86**	**37.89**	**27.04**	**57.33**
Scan2Cap	SCST	-	-	-	-	-	-	-	-	-	-	-	-	48.38	26.09	22.15	44.74
D3Net		-	-	-	-	-	-	-	-	-	-	-	-	62.64	35.68	25.72	53.90
χ-Tran2Cap		58.81	34.17	25.81	54.10	41.52	23.83	21.90	44.97	61.83	35.65	26.61	54.70	43.87	25.05	22.46	45.28
Contextual		-	-	-	-	50.29	25.64	22.57	44.71	-	-	-	-	54.30	27.24	23.30	45.81
Vote2Cap-DETR		84.15	42.51	28.47	59.26	73.77	38.21	26.64	54.71	86.28	42.64	28.27	59.07	70.63	35.69	25.51	52.28
Ours		**89.72**	**44.56**	**28.96**	**62.13**	**83.14**	**42.17**	**27.92**	**59.44**	**89.71**	**45.31**	**29.06**	**62.11**	**79.84**	**40.84**	**27.28**	**57.54**

Table 1: Experimental results on the ScanRefer (Chen et al., 2020a). C, B-4, M, and R represent the captioning metrics CIDEr (Vedantam et al., 2015), BLEU-4 (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), and ROUGE-L (Chin-Yew, 2004), respectively. A higher score for each indicates better performance.

4 Experiments

4.1 Datasets and Metrics

Datasets. In our studies, we focus on 3D dense captioning and employ two established datasets: ScanRefer (Chen et al., 2020a) and Nr3D (Achlioptas et al., 2020). These datasets are rich in human-generated descriptions, with ScanRefer providing $36,665$ descriptions for $7,875$ objects across 562 scenes, and Nr3D offering $32,919$ descriptions for $4,664$ objects in 511 scenes. For training, these descriptions and objects are derived from the ScanNet (Dai et al.) database, which comprises $1,201$ 3D scenes. For evaluation, we use $9,508$ descriptions from ScanRefer and $8,584$ from Nr3D, corresponding to $2,068$ and $1,214$ objects across 141 and 130 scenes, respectively, from the 312 3D scenes in the ScanNet validation set.

Metrics. We evaluate the model using four types of performance metrics: CIDEr (Vedantam et al., 2015), BLEU-4 (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), and ROUGE-L (Chin-Yew, 2004), denoted as **C**, **B-4**, **M**, and **R**, respectively. Following the previous studies (Chen et al., 2021; Cai et al., 2022; Jiao et al., 2022; Wang et al., 2022; Chen et al., 2023a), we first employ Non-Maximum Suppression (NMS) to eliminate redundant object predictions from the object proposals. Each proposal is represented as a pair consisting of a predicted bounding box $\hat{b}_i$ and a generated caption $\hat{c}_i$. To assess both the model's ability to locate objects and generate captions accurately, we employ the $m@k$, setting the IoU threshold k at 0.25 and 0.5 for our experiments, following (Chen et al., 2021):

$$m@k = \frac{1}{N}\sum_{i=1}^{N} m\left(\hat{c}_i, C_i\right) \cdot \mathbb{I}\left\{\mathrm{IoU}\left(\hat{b}_i, b_i\right) \geq k\right\}, \tag{9}$$

where N is the number of all annotated instances in the evaluation set, and m represents the captioning metrics C, B-4, M, and R.

4.2 Implementation Details

Our training phase is structured into three stages, following the approach of (Chen et al., 2023a). Initially, we pre-train our network excluding the caption head on the ScanNet (Dai et al.) dataset for $1,080$ epochs. The batch size is 8. The loss function is minimized using AdamW optimizer (Loshchilov and Hutter, 2017), for which the initial learning rate is 5×10^{-4} and decreases to 10^{-6} according to a cosine annealing schedule. We also apply a weight decay of 0.1 and a gradient clipping of 0.1, as suggested by (Misra et al., 2021). Afterward, starting from the pre-trained weights, we jointly train the entire model with the standard cross-entropy loss for an additional 720 epochs on the ScanRefer (Chen et al., 2020a) and Nr3D (Achlioptas et al., 2020) datasets, fixing the detector's learning rate at 10^{-6} and reducing the caption head's from 10^{-4} to 10^{-6} to prevent over-fitting (about 20/17 hours for ScanRefer/Nr3D). In the SCST (Rennie et al., 2017) phase, we adjust the caption head using a batch size of 2 while keeping the detector fixed

Model	Training	C@0.5 ↑	B-4@0.5 ↑	M@0.5 ↑	R@0.5 ↑
Scan2Cap		27.47	17.24	21.80	49.06
D3Net		33.85	20.70	23.13	53.38
SpaCap3d		33.71	19.92	22.61	50.50
3DJCG		38.06	22.82	23.77	52.99
Contextual	MLE	35.26	20.42	22.77	50.78
REMAN		34.81	20.37	23.01	50.99
Vote2Cap-DETR		43.84	26.68	25.41	54.43
Ours		**56.39**	**30.87**	**27.54**	**60.36**
D3Net		38.42	22.22	24.74	54.37
χ-Tran2Cap		33.62	19.29	22.27	50.00
Contextual	SCST	37.37	20.96	22.89	51.11
Vote2Cap-DETR		45.53	26.88	25.43	54.76
Ours		**59.48**	**32.60**	**27.99**	**61.08**

Table 2: Experimental results on the Nr3D (Achlioptas et al., 2020) with IoU threshold at 0.5.

over a span of 180 epochs and maintain a constant learning rate of 10^{-6} (about 22/18 hours for ScanRefer/Nr3D). In the experimental setup that uses additional 2D data, as shown in Table 1, we employ the pre-trained ENet (Chen et al., 2020b) to extract 128-dimensional multiview features from 2D ScanNet images, as in the (Chen et al., 2021). The parameter size of our model is 21M, and the average inference time on the evaluation set of the ScanRefer is 1.8ms. All experiments of our SIA are conducted with one Titan RTX GPU on PyTorch (Paszke et al., 2019).

4.3 Comparison with Existing Methods

In this section, we benchmark our performance against eleven state-of-the-art methods: Scan2Cap (Chen et al., 2021), D3Net (Chen et al., 2022), SpaCap3D (Wang et al., 2022), MORE (Jiao et al., 2022), 3DJCG (Cai et al., 2022), Contextual (Zhong et al., 2022), REMAN (Mao et al., 2023), 3D-VLP (Jin et al., 2023), χ-Tran2Cap (Yuan et al., 2022), Vote2Cap-DETR (Chen et al., 2023a), and Unit3D (Chen et al., 2023b). We apply IoU thresholds of 0.25 and 0.5 for ScanRefer (Chen et al., 2020a) as shown in Table 1 and an IoU threshold of 0.5 for Nr3D (Achlioptas et al., 2020) indicated in Table 2. For the baselines, we present the evaluation results reported in the original papers, and "-" in Table 1 and Table 2 means that such results have not reported in the original paper or follow-up study.

ScanRefer. The descriptions in the ScanRefer include depictions of the target object's attributes and information about the spatial relationships between this target object and other surrounding objects. Table 1 summarizes the results on the ScanRefer dataset. Our method significantly surpasses current methods in 3D dense captioning across all input data settings and IoU threshold configurations. We attribute this enhancement to our contextualized late aggregation mechanism. Unlike previous object-centric methods where surrounding information is bound to the center of object proposals, SIA directly targets the region-of-interest for contextual captions with the context query while simultaneously addressing object localization and its attribute captions with the instance query.

Model	IoU=0.50					
	C↑	B-4↑	M↑	R↑	mAP↑	AR↑
Vote2Cap-DETR	73.77	38.21	26.64	54.71	45.56	67.77
SIA using only V^o	73.90	40.67	26.76	55.31	48.09	68.43
SIA w/o V^g	81.45	41.19	26.33	56.71	48.74	68.13
SIA (Ours)	**83.14**	**42.17**	**27.92**	**59.44**	**49.69**	**69.08**

Table 3: Ablation study on the ScanRefer (Chen et al., 2020a). The core components of SIA: i) decomposing the query set into the instance query V^o and the context query V^c, ii) generating the global feature V^g, and iii) aggregating the TGI feature V^a.

Method	C@0.5 ↑	B-4@0.5 ↑	M@0.5 ↑	R@0.5 ↑
Contexts V^c	72.82	38.46	26.76	56.71
Single Instance V_i^o & Contexts V^c	72.89	38.21	27.04	57.33
Instances V^o & Contexts V^c (Ours)	**73.22**	**40.91**	**28.19**	**60.46**

Table 4: Ablation for how the instance feature V^o and the context features V^c participate in the Global Aggregator $G(\cdot)$ on the ScanRefer (Chen et al., 2020a). $\cdot_i$ denotes a single i-th decoded query feature.

Nr3D. The Nr3D dataset is designed to evaluate the model's performance in interpreting free-form natural language descriptions of objects as spoken by humans. Our SIA quantitatively demonstrates its ability to generate various descriptions for an object by showing state-of-the-art performance across all evaluation metrics, as shown in Table 2.

4.4 Ablation Study and Discussion

The core components of SIA consist of three factors: i) decomposing the query set into instance query V^o and context query V^c, ii) generating the global descriptor V^g, and iii) composing the fully informed contextualized feature V^a using our TGI-Aggregator. In our ablation study, we validate that every component of our proposed SIA positively contributes to the final performance.

Instance Query Generator. We define SIA using only the Instance Query Generator (i.e., SIA using only V^o in Table 3) as our baseline and compare it with Vote2Cap-DETR (Chen et al., 2023a), an object-centric transformer encoder-decoder ar-

	C@0.5 ↑	B-4@0.5 ↑	M@0.5 ↑	R@0.5 ↑
SIA with GeM Pooling	66.97	36.97	26.76	56.71
SIA with NetVLAD (Ours)	**73.22**	**40.91**	**28.19**	**60.46**

Table 5: Experimental results comparing Global Aggregators on the ScanRefer (Chen et al., 2020a).

	C@0.5 ↑	B-4@0.5 ↑	M@0.5 ↑	R@0.5 ↑
SIA with K=8	69.86	37.89	27.04	57.33
SIA with K=16 (Ours)	73.22	**40.91**	**28.19**	**60.46**
SIA with K=32	**73.38**	37.92	27.92	60.16

Table 6: Performance variation according to the size of K on the ScanRefer (Chen et al., 2020a).

Object	scene0144_00 | 20 | nightstand
Only use V^o	"this is a brown nightstand ."
Only use V^a	"it is to the left of the bed ."
SIA	"this is a brown nightstand . it is to the left of the bed ."
GT	"there is a nightstand on the wall . it is to the left of a bed ."
Object	scene0019_00 | 9 | vending_machine
Only use V^o	"the vending machine is a white rectangle ."
Only use V^a	"the vending machine is in the corner of the room ."
SIA	"the vending machine is in the corner of the room . the vending machine is a white rectangle ."
GT	"this is a vending machine . it is in the corner of the room , by a lamp ."

Table 7: Qualitative results on the ScanRefer (Chen et al., 2020a) generated from i) using only instance features V^o, ii) using the fully informed contextualized feature V^a from the TGI-Aggregator, and iii) concatenating both captions for the final caption of SIA.

chitecture. The major difference between our baseline and Vote2Cap-DETR is how we generate the query set for instances. Vote2Cap-DETR uses farthest point sampling (FPS) to generate queries before the query coordinates are adjusted through voting. Therefore, if the coordinates are mistakenly focused on a specific object after voting, features will be extracted from the same object. On the other hand, our baseline extracts the features from the candidate coordinates after the voting. This enhancement boosts localization performance in terms of mean Average Precision (mAP) and Average Recall (AR), naturally leading to improvements in dense captioning performance.

Context Query Generator. SIA decomposes the role of queries into instance query V^o that focuses on the object itself and context query V^c that designates the contextual region. SIA w/o V^g in Table 3 shows the result of generating a caption using only context query V^c and instance query V^o in the TGI-aggregator, excluding the global descriptor. By showing high performance improvement compared to the results of SIA using only V^o, we demonstrate the effectiveness of query set separation.

TGI-Aggregator. As shown in Table 3, utilizing a global descriptor V^g in the TGI-Aggregator results in performance improvements across all aspects. Table 4 shows our ablation study on how we aggregate the decoded context query V^c and instance query V^o to construct the global feature V^g that is afterward fed to our TGI-Aggregator (recall Figure 3). The scenarios include i) aggregating all context features V^c, ii) gathering one instance feature V_i^o with all context features V^c, and iii) aggregating all instance features V^o and context features V^c to extract a global feature. Aggregating all instance and context features to create a global feature results in the best performance. This implies that reflecting all instances and contexts is better when representing the entire scene.

We also compare two of the most widely used aggregation frameworks for whole-scene representation: GeM pooling (Radenović et al., 2019) and NetVLAD (Arandjelovic et al., 2016), as shown in Table 5. We empirically adopt NetVLAD for our Global Aggregator. In our TGI-Aggregator, we concatenate the global descriptor V^g to each decoded instance query V_i^o and K context features that are locationally closest to V_i^o within V^c.

Analysis of hyper-parameter K. In the TGI-Aggregator, we concatenate the global descriptor V^g to each decoded instance query V_i^o and K nearest features within V^c in terms of spatial proximity to V_i^o. Based on the experimental results of Table 6, we set $K = 16$. A performance decrease is observed with $K = 8$, likely due to insufficient contextual information, while $K = 32$ shows little performance change despite significantly increasing memory and execution time costs.

4.5 Qualitative Analysis

We qualitatively present the captions generated from i) using only instance features V^o, ii) using only the fully informed contextualized feature V^a from the TGI-Aggregator, as shown in Table 7. It can be seen that while the captions generated from V^o include descriptions for attributes, the captions generated from V^a include contextual information such as relations with other objects and the global scene.

We also provide a qualitative comparison

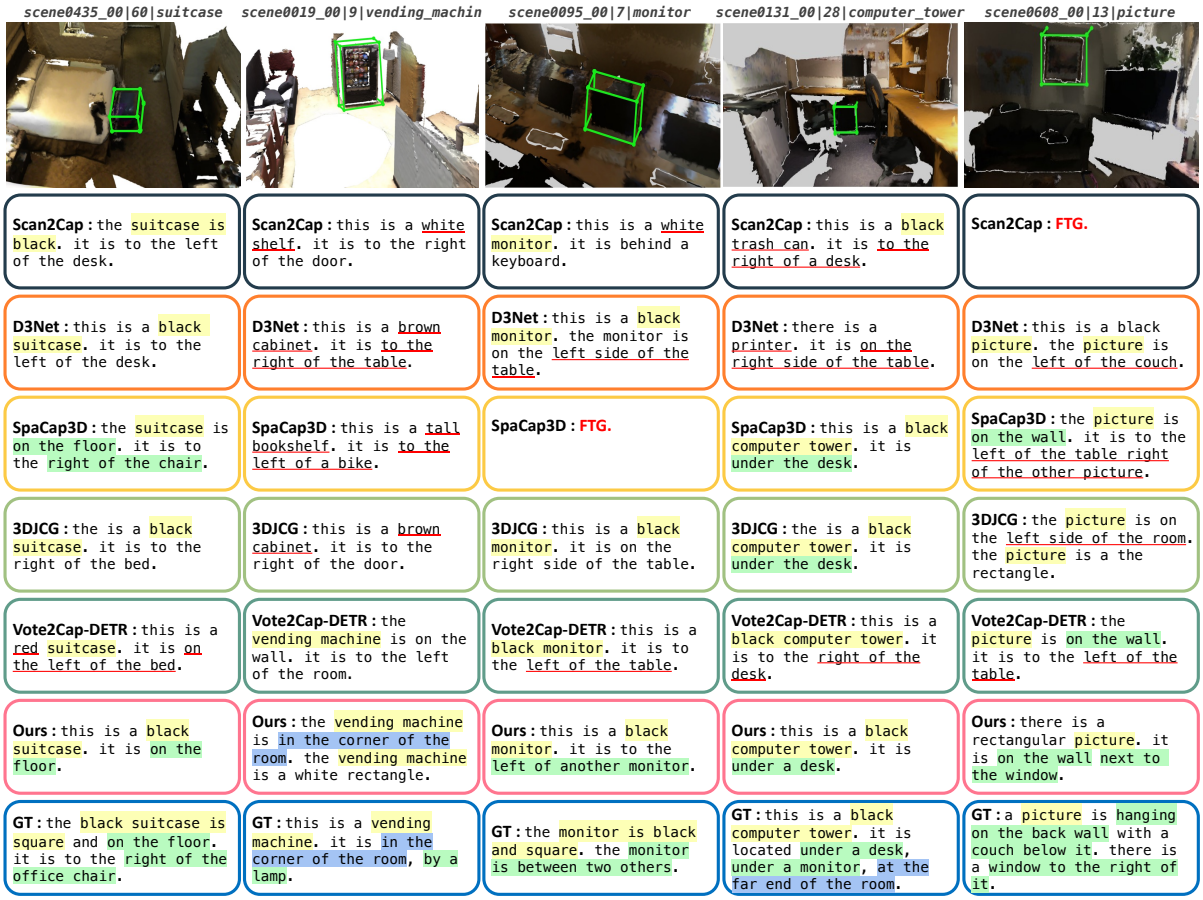

Figure 4: Qualitative results on the ScanRefer (Chen et al., 2020a). The yellow-highlighted sections show information specific to the object itself, the green-highlighted sections describes the relationships between objects, and the blue-highlighted sections depict the spatial position of the object in the 3D scene. Captions underlined in red indicate incorrect descriptions. **FTG.** represent failures in caption generation due to low IoU.

with the state-of-the-art models: Scan2Cap (Chen et al., 2021), SpaCap3D (Wang et al., 2022), D3Net (Chen et al., 2022), 3DJCG (Cai et al., 2022), and Vote2Cap-DETR (Chen et al., 2023a). The ground-truth includes descriptions of the intrinsic properties of the object (e.g., the monitor is black and square.), explanations using the relationships between objects (e.g., a picture is hanging on the back wall with a couch below it.), and descriptions of the object in the context of the entire space (e.g., at the far end of the room.). Existing models, focusing on objects, generate captions limited to the object and its immediate relations in a fixed format (e.g., this is a black suitcase.). Our model can handle not only object queries but also context queries, allowing it to generate sentences in various formats (e.g., the vending machine is a white rectangle.) and create descriptions of the entire space (e.g., the vending machine is in the corner of the room.). These results emphasize the importance of an integrated understanding of the object, its surroundings, and the overall space in captioning.

5 Conclusion

In the 3D dense captioning task, the description of an object within a 3D scene encompasses not only the intrinsic characteristics of the object but also the relationship with surrounding objects and the spatial relationship of the object with respect to the overall space. In this work, we propose a novel approach that independently generates captions with different region of interests and aggregates them afterwards to enhance local-global sensitivity of descriptions. Through extensive experiments on benchmark datasets, our method significantly improves 3D dense captioning over previous approaches, demonstrating the importance of an integrated understanding of objects, surroundings, and overall space for caption generation.

Limitations

SIA still faces the limitations of prior set prediction architectures, where the number of instance and context queries must be heuristically determined. Future work could explore methods that allow for the dynamic adjustment of instances and context query numbers based on the complexity of the scene.

Ethical Considerations

3D dense captioning is the task of locating objects in a 3D scene and generating captions for each object. Our proposed method, specifically designed for this task, is ethically sound, employing only publicly available datasets throughout our research. These benchmark datasets feature 3D indoor environments exclusively populated with objects.

Acknowledgement

We sincerely thank Junhyug Noh, Seokhee Hong, Jaewoo Ahn, and Dayoon Ko for their constructive comments. This work was supported by LG AI Research and Institute of Information & Communications Technology Planning & Evaluation (IITP) grant (No. RS-2019-II191082, No. RS-2022-II220156) funded by the Korea government (MSIT).

References

Panos Achlioptas, Ahmed Abdelreheem, Fei Xia, Mohamed Elhoseiny, and Leonidas Guibas. 2020. Referit3d: Neural listeners for fine-grained 3d object identification in real-world scenes. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 422–440. Springer.

Relja Arandjelovic, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. 2016. Netvlad: Cnn architecture for weakly supervised place recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5297–5307.

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Daigang Cai, Lichen Zhao, Jing Zhang, Lu Sheng, and Dong Xu. 2022. 3djcg: A unified framework for joint dense captioning and visual grounding on 3d point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16464–16473.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer.

Dave Zhenyu Chen, Angel X Chang, and Matthias Nießner. 2020a. Scanrefer: 3d object localization in rgb-d scans using natural language. In *European conference on computer vision*, pages 202–221. Springer.

Dave Zhenyu Chen, Qirui Wu, Matthias Nießner, and Angel X Chang. 2022. D3net: A unified speaker-listener architecture for 3d dense captioning and visual grounding. In *European Conference on Computer Vision*, pages 487–505. Springer.

Jintai Chen, Biwen Lei, Qingyu Song, Haochao Ying, Danny Z Chen, and Jian Wu. 2020b. A hierarchical graph network for 3d object detection on point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 392–401.

Sijin Chen, Hongyuan Zhu, Xin Chen, Yinjie Lei, Gang Yu, and Tao Chen. 2023a. End-to-end 3d dense captioning with vote2cap-detr. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11124–11133.

Zhenyu Chen, Ali Gholami, Matthias Nießner, and Angel X Chang. 2021. Scan2cap: Context-aware dense captioning in rgb-d scans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3193–3203.

Zhenyu Chen, Ronghang Hu, Xinlei Chen, Matthias Nießner, and Angel X Chang. 2023b. Unit3d: A unified transformer for 3d dense captioning and visual grounding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18109–18119.

Lin Chin-Yew. 2004. Rouge: A package for automatic evaluation of summaries. In *Proceedings of the Workshop on Text Summarization Branches Out, 2004*.

Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Yang Jiao, Shaoxiang Chen, Zequn Jie, Jingjing Chen, Lin Ma, and Yu-Gang Jiang. 2022. More: Multi-order relation mining for dense captioning in 3d scenes. In *European Conference on Computer Vision*, pages 528–545. Springer.

Zhao Jin, Munawar Hayat, Yuwei Yang, Yulan Guo, and Yinjie Lei. 2023. Context-aware alignment and mutual masking for 3d-language pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10984–10994.

Harold W Kuhn. 1955. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Aihua Mao, Zhi Yang, Wanxin Chen, Ran Yi, and Yongjin Liu. 2023. Complete 3d relationships extraction modality alignment network for 3d dense captioning. *IEEE Transactions on Visualization and Computer Graphics*.

Ishan Misra, Rohit Girdhar, and Armand Joulin. 2021. An end-to-end transformer model for 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2906–2917.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Charles R Qi, Or Litany, Kaiming He, and Leonidas J Guibas. 2019. Deep hough voting for 3d object detection in point clouds. In *proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9277–9286.

Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. 2017. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30.

F. Radenović, G. Tolias, and O. Chum. 2019. Fine-tuning cnn image retrieval with no human annotation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(7):1655–1668.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. Self-critical sequence training for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7008–7024.

Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4566–4575.

Heng Wang, Chaoyi Zhang, Jianhui Yu, and Weidong Cai. 2022. Spatiality-guided transformer for 3d dense captioning on point clouds. *arXiv preprint arXiv:2204.10688*.

Zhihao Yuan, Xu Yan, Yinghong Liao, Yao Guo, Guanbin Li, Shuguang Cui, and Zhen Li. 2022. X-trans2cap: Cross-modal knowledge transfer using transformer for 3d dense captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8563–8573.

Yufeng Zhong, Long Xu, Jiebo Luo, and Lin Ma. 2022. Contextual modeling for 3d dense captioning on point clouds. *arXiv preprint arXiv:2210.03925*.