POP−3D: Open-Vocabulary 3D Occupancy Prediction from Images

Antonin Vobecky[1,2] Oriane Siméoni[2] David Hurych[2] Spyros Gidaris[2]
Andrei Bursuc[2] Patrick Pérez[2] Josef Sivic[1]
[1]CTU CIIRC, [2]valeo.ai

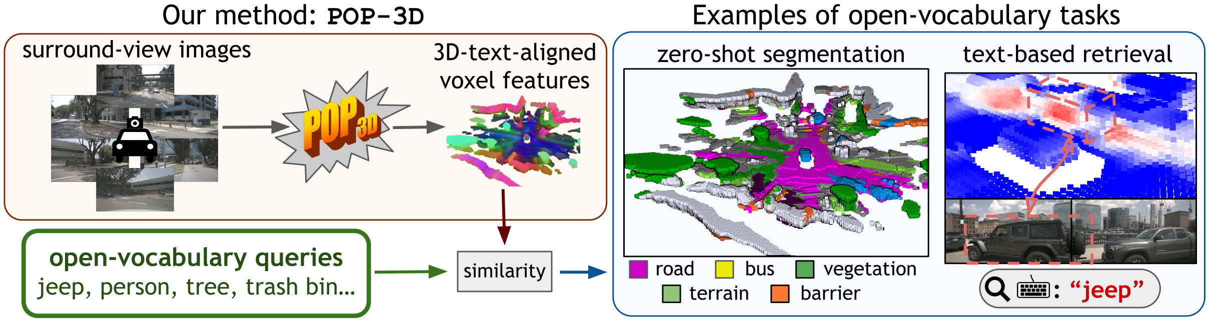


Figure 1: **Overview.** Given surround-view images, POP−3D produces a voxel grid of text-aligned features that support open-vocabulary downstream tasks such as zero-shot occupancy segmentation or text-based grounding and retrieval.

Abstract

We propose an approach to predict a 3D semantic voxel occupancy map from input 2D images with features allowing 3D grounding, segmentation and retrieval of free-form language queries. To this end: We design a new architecture that consists of a 2D-3D encoder together with occupancy prediction and 3D-language heads; We develop a tri-modal *self-supervised training that leverages three modalities – images, language and LiDAR point clouds– and enables learning the proposed architecture using a strong pre-trained vision-language model without the need for any 3D manual annotations. We quantitatively evaluate the proposed model on the task of zero-shot 3D semantic segmentation using existing datasets and show results on the tasks of 3D grounding and retrieval of free-form language queries.*

1. Introduction

The detailed analysis of 3D environments –both geometrically and semantically– is a fundamental perception task in many applications. It is usually conducted with cameras and/or laser scanners (LiDAR). In its most complete version, called *semantic 3D occupancy* prediction, it amounts to labelling each voxel of the perceived volume as occupied by a certain class of objects or empty. This is challenging since sensors only capture information about visible surfaces. Recent works, e.g., [9], leverage manually-annotated LiDAR data to produce partial annotation of the 3D occupancy space. However, such an annotation remains difficult to scale, even with sparse point clouds, and limits the learned representation to encoding a closed vocabulary, i.e., a limited predefined set of classes. We tackle those challenges and propose an open-vocabulary approach to 3D semantic occupancy prediction that relies only on unlabeled image-LiDAR data for training and pre-trained image-language model. In addition, our model uses only camera inputs at run time, bypassing altogether the need for expensive dense LiDAR sensor, in contrast with most 3D semantic perception systems (whether at point or voxel level).

In this work, we attack the difficult problem of 3D semantic occupancy prediction, and leverage the progress made recently in supervised 3D occupancy prediction [9] and in language-image alignment [18]. We named our approach POP−3D (for o**P**en-vocabulary **O**ccupancy **P**rediction in **3D**). Underneath is a two-head image-only model trained with aligned image-LiDAR raw data, meaning that we use no manual annotations. First, we train a class-agnostic occupancy prediction head supervised using the sparse LiDAR scans. We additionally train the model to predict open-vocabulary features by producing LiDAR-image alignment supervision. At inference, the 3D-occupancy features can be prompted to obtain open-vocabulary segmentation.

2. Related work

Recent works produce dense semantic occupancy predictions from a single image by projecting image features into 3D voxels [4], by exploiting tri-perspective view representations [5] augmenting the standard BEV with two additional perpendicular planes to recover the full 3D [9]. In contrast, we do not need human-made annotation and produce semantic 3D occupancy predictions using supervision from LiDAR and from an image-language model allowing our model to acquire open-vocabulary skills in the voxel space.

Image-language aligned models project images and text into a shared representation space [6, 7, 11, 14–17]. Contrastive image-language learning on many millions of image-text pairs [11, 16] leads to high-quality representations with impressive zero-shot skills from one modality to the other. We use CLIP [16] for its appealing open-vocabulary property that enables the querying of visual content with natural language toward recognizing objects of interest without manual labels. `POP-3D` uses LiDAR supervision for precise occupancy prediction and learns to produce in the 3D space CLIP-like features easily paired with language.

CLIP features can be projected into 3D meshes [10] and NeRFs [12] to enable language queries. Originally producing image-level embeddings, CLIP can be extended to pixel-level predictions for open-vocabulary semantic segmentation, e.g., by MaskCLIP [18]. It adjusts the attentive-pooling layer of CLIP to generate pixel-level CLIP features that are distilled into an encoder-decoder semantic segmentation network. We exploit MaskCLIP+ [18] in our approach, and generate target 3D CLIP-like features by mapping MaskCLIP+ pixel-level features to LiDAR points observed in images.

3. Open-vocabulary 3D occupancy prediction

3.1. Architecture

Given a set of surround-view images captured from one world location, our goal is to output a 3D occupancy voxel map and to support language-driven tasks. To reach these goals, we propose an architecture composed of three modules (Fig. 2(a)). First, a *2D-3D encoder* predicts a voxel feature grid from the input images. Second, the *occupancy head* predicts for the entire voxel grid which voxels are free and which are occupied. Finally, the *3D-language head* is applied on each occupied voxel to output a language embedding vector enabling a range of 3D open-vocabulary tasks. The three modules are described next.

2D-to-3D encoder f_{3D}. Given surround-view camera RGB images $\mathbf{I}$, the encoder f_{3D} produces a feature voxel grid $\mathbf{V} = f_{\text{3D}}(\mathbf{I}) \in \mathbb{R}^{H_\text{V} \times W_\text{V} \times D_\text{V} \times C_\text{V}}$, where H_V, W_V and D_V are the spatial dimensions of the voxel grid, and C_V is the feature dimension of each voxel. This feature voxel grid is then passed to two prediction heads described next.

Occupancy head g. Given the feature voxel grid $\mathbf{V}$, the occupancy prediction head g aims at classifying every voxel as 'empty' or 'occupied'. Following [9], this head is implemented as a non-linear network composed of N_occ hidden blocks with configuration `Linear-Softplus-Linear`, each with $C_\text{occ}^\text{hidden}$ hidden features, and a final linear classifier outputting two logits, one per class. It outputs the tensor

$$\mathbf{O}_\text{occ} = g\left(\mathbf{V}\right) \in \mathbb{R}^{H_\text{V} \times W_\text{V} \times D_\text{V} \times 2}, \tag{1}$$

containing the occupancy prediction for each voxel.

3D language head h. In parallel, the voxel grid $\mathbf{V}$ is fed to a language feature extractor. This head processes voxel features to output embedding vectors that are aligned to vision-language representations, such as CLIP [16], aiming to inherit their open-vocabulary abilities, i.e., enabling 3D language-driven tasks such as zero-shot 3D semantic segmentation. This head has the same architecture as the occupancy one, just with $C_\text{ft}^\text{hidden}$ hidden features, and a final linear layer that outputs C_ft^out-dimensional vision language embedding for each voxel. It outputs the tensor

$$\mathbf{O}_\text{ft} = h\left(\mathbf{V}\right) \in \mathbb{R}^{H_\text{V} \times W_\text{V} \times D_\text{V} \times C_\text{ft}^\text{out}}, \tag{2}$$

containing the predicted vision-language embeddings.

3.2. Tri-modal self-supervised training

We propose a tri-modal self-supervised learning algorithm that leverages three modalities: (i) images, (ii) language and (iii) LiDAR point clouds. The training algorithm is illustrated in Fig. 2(b). The training is implemented via two losses that are used to train the two heads of the proposed architectures jointly with the 2D-to-3D encoder.

Occupancy loss. We guide the occupancy head g to perform a class-agnostic occupancy prediction by the available unlabeled LiDAR point clouds, which we convert to occupancy prediction targets $T_\text{occ} \in \{0, 1\}$. Each voxel location x containing at least one LiDAR point is labeled as 'occupied' (i.e., $T_\text{occ}(x) = 1$) and as 'empty' otherwise ($T_\text{occ}(x) = 0$). Having these targets, we supervise the occupancy prediction head densely at all locations of the voxel grid. The occupancy loss $\mathcal{L}_\text{occ}$ is a combination of cross-entropy loss $\mathcal{L}_\text{CE}$ and Lovász-softmax [1] loss $\mathcal{L}_\text{Lov}$ between the predicted occupancy tensor $\mathbf{O}_\text{occ}$ and the occupancy targets tensor $\mathbf{T}_\text{occ}$.

Image-language distillation. We supervise the 3D-language head at the level of points $p_n \in P_\text{cam}$ which project to at least one of the cameras, i.e., $P_\text{cam} \subset P$, where P is the complete point cloud. To get a feature target for a 3D point $p_n \in P_\text{cam}$ in the voxel feature grid, we use the camera projection function Π_c that projects 3D point p_n into 2D coordinates $u_n = (u_n^{(x)}, u_n^{(y)})$ in camera c. To to obtain feature targets $\mathbf{T}_\text{ft}$ for 3D points in P_cam with corresponding 2D projections $U = \{\Pi_c\left(p_n\right)\}_{n=1}^{N}$ in camera c, we run the language-image-aligned feature extractor f_I on image $\mathbf{I}_c$, and use the 2D projections' coordinates to sample from the resulting feature map, i.e., $\text{T}_\text{ft} = \{f_\text{I}\left(\mathbf{I}_c\right)[u_n^{(x)}, u_n^{(y)}]\}_{n=1}^{N} \in \mathbb{R}^{N \times C_\text{ft}^\text{out}}$, where $[x, y]$ is an indexing operator in the extracted feature map. To train the 3D language head, we use L_2 mean

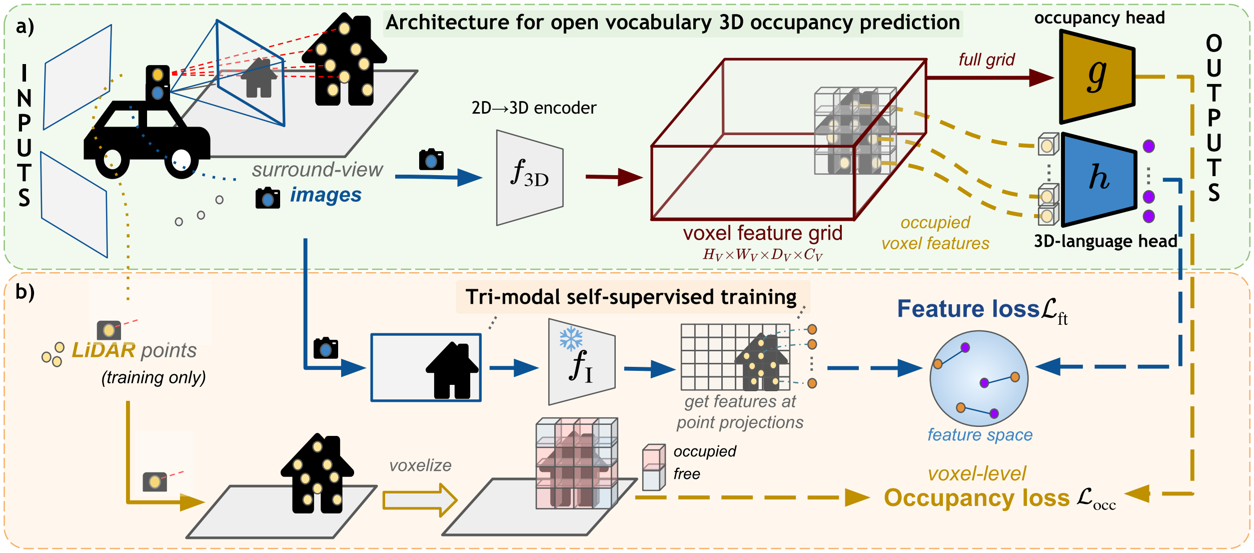


Figure 2: **Approach. (a)** With surround-view images on the input, the model extracts a voxel feature grid that is fed to two parallel heads: an occupancy head g, and a 3D-language feature head h. **b)** *Training*. The occupancy loss $\mathcal{L}_{\text{occ}}$ is used to train class-agnostic occupancy predictions. The feature loss $\mathcal{L}_{\text{ft}}$ enforces the 3D-language head h to output text-aligned features.

squared error loss $\mathcal{L}_{\text{ft}}$ between the targets $\mathbf{T}_{\text{ft}}$ and the predicted features $\tilde{\mathbf{O}}_{\text{ft}} \in \mathbb{R}^{N \times C_{\text{ft}}^{\text{out}}}$.

The final loss sums the *occupancy* and *image-language* losses $\mathcal{L} = \mathcal{L}_{\text{occ}} + \lambda \mathcal{L}_{\text{ft}}$, with λ balancing the two terms.

3.3. 3D open-vocabulary test-time inference

We focus on two downstream tasks: (i) zero-shot 3D semantic segmentation and (ii) language-driven 3D grounding.

Zero-shot 3D semantic segmentation from images. Unlike supervised approaches that necessitate retraining when the set of target classes changes, our approach requires training the model only once. We can adjust the number of segmented classes effortlessly by providing a different set of input text queries. In detail, at test-time, we first feed a set of test surround-view images $\mathbf{I}$ into the trained `POP-3D` network, resulting in class-agnostic occupancy prediction $\mathbf{O}_{\text{occ}}$ via the occupancy head g, and language-aligned feature predictions $\mathbf{O}_{\text{ft}}$ via the 3D-language head h. Next, to obtain text features for the input queries, we follow the same strategy as [8]. Finally, considering M text features, one for each of the M target segmentation classes, we measure their similarity to the predicted language-aligned features $\mathbf{O}_{\text{ft}}$ at occupied voxels obtained from $\mathbf{O}_{\text{occ}}$. We assign the label with the highest similarity to each occupied voxel.

Language-driven 3D grounding. The task of language-driven 3D grounding is performed in a similar manner. However, here only a single input language query is given. Once determined the occupied voxels from $\mathbf{O}_{\text{occ}}$, we compute the similarity between the encoded input text query and predicted features $\mathbf{O}_{\text{ft}}$ at the occupied voxels. The resulting similarity score can be visualized as a heat-map, as shown in Fig. 1, or thresholded to obtain the location of the target.

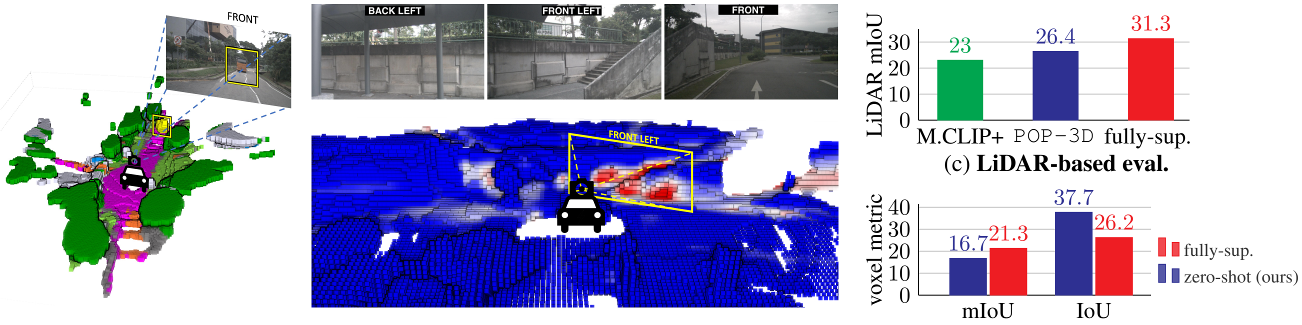
4. Experiments

Dataset. We use the nuScenes [3] dataset, which provides 3D point clouds, surround-view images obtained from six cameras mounted at the top of the car, and projection matrices between the 3D point cloud and cameras.

Metrics. We measure the class-dependent mean Intersection over Union (mIoU) and the class-agnostic occupancy Intersection over Union (IoU).

New evaluation protocol for 3D occupancy prediction. The task of 3D occupancy prediction has no established evaluation protocol yet. TPVFormer [9] did not introduce any evaluation protocol. Since voxel semantic segmentation consists of both *occupancy prediction* of the voxel grid and *classification of occupied voxels*, it is not enough to evaluate just at the points of ground-truth information from the LiDAR, as this does not take free space prediction into account. To tackle this, we take inspiration from [2] and propose to obtain the evaluation labels from the available LiDAR point clouds. First, LiDAR rays passing through 3D space set the labels of intersected voxels to *free*. Second, voxels containing LiDAR points are assigned the most frequent semantic label of points lying within (or an *occupied label* in the case of class-agnostic evaluation). Third, all other voxels are *ignored* during evaluation, as they were not observed by any LiDAR ray, and we are not certain if they are occupied.

Implementation details. We use the recent TPVFormer [9] with ResNet-101 image backbone as the 2D-3D encoder $f_{3\text{D}}$. For the language-image feature extractor, we use MaskCLIP + [18], which provides features of dimension $C_{\text{ft}}^{\text{out}} = 512$. We use the default learning rate (LR) of 2e-4, Adam [13] optimizer, a cosine learning rate scheduler with



(a) **Zero-shot segmentation**

(b) **Retrieval** with query "*stairs*"

(c) **LiDAR-based eval.**

(d) **3D occupancy pred.**

Figure 3: (a) **Zero-shot semantic 3D occupancy prediction** on the 16 classes in the nuScenes [3] val. split. Note how our method is able to segment objects in 3D, such as a bus (yellow), from only input 2D images and in a zero-shot manner. Visualizations are shown on an interpolated 300x300x24 grid. (b) **Text-based retrieval** with query '*stairs*'. Red means a high similarity of the text query to the 3D features. **Comparison to SoTA**: (c) We compare to baselines following the LiDAR-based evaluation and (d) our proposed occupancy evaluation.

final LR 1e-6, and with linear warmup from 1e-5 LR for the first 500 iterations. Both prediction heads have two layers, i.e., $N_{\text{occ}} = N_{\text{ft}} = 2$, and $C_{\text{occ}} = 512$ and $C_{\text{ft}} = 1024$ feature channels. We train our model for 12 epochs. We put the same weight to the occupancy and feature losses, i.e., $\lambda = 1$.

Illustration of open-vocabulary capabilities In Fig. 3b we visualize text-based 3D object retrievals inside a scene using the query 'stairs'. The results show that our model is able to localize in 3D space fine-grained language queries.

4.1. Comparison to state of the art

Here we compare our approach to two state-of-the-art methods: (i) fully supervised (closed-vocabulary) TPVFormer [9] and open-vocabulary image-based MaskCLIP+ [18] backprojected to 3D via LiDAR.

Comparison to a fully-supervised TPVFormer [9]. In. Fig. 3d, we compare to the supervised TPVFormer [9] in terms of class-agnostic IoU and (16+1)-class mIoU (16 semantic classes plus the *empty* class) on the nuScenes [3] val. set. Interestingly, our model outpeforms its supervised counterpart in the class-agnostic IoU by 11.5 points, showing superiority in the prediction of the occupied space. This can be attributed to different training schemes: in the fully-supervised case, the *empty* class competes with the other semantic classes, whereas in our case the occupancy head performs only class-agnostic occupancy prediction. Next, for the semantic occupancy segmentation, we see that our zero-shot approach reaches $\approx 78\%$ (16.7 vs. 21.3 mIoU) of the supervised model performance, which we consider as strong result given that the latter requires manually-annotated point clouds for training. In contrast, our approach is zero-shot and does not require any manual point cloud annotations at training. We show qualitative results of POP-3D in Fig. 3a.

Comparison to MaskCLIP+ [18]. In Fig. 3c we compare the quality of the 3D vision-language features learnt by our POP-3D against the strong *MaskCLIP+* [18] baseline. We project the 3D LiDAR points to the 2D image(s) space, sample MaskCLIP+ [18] features extracted from the 2D image, and backproject them to 3D via the LiDAR rays. For a fair comparison, we evaluate only the LiDAR points with a projection to the camera (LiDAR mIoU), i.e., this evaluation considers only the classification of occupied points in space, not the occupancy prediction itself. We observe that POP-3D outperforms MaskCLIP+ (26.4 vs. 23.0 mIoU), i.e., our method manages to learn better 3D vision-language features than its teacher, while also not requiring LiDAR data at test time (as MaskCLIP+ does). We attribute this to a slight adaptation of the features to the nuScenes dataset and to the smoothing of the features during the training. Finally, in Fig. 3c, we see that POP-3D reaches $\approx 84\%$ of the fully-supervised model [9]'s LiDAR mIoU performance.

5. Conclusion

In this paper we propose POP-3D, a tri-modal self-supervised learning strategy with a novel architecture that enables open-vocabulary voxel segmentation from 2D images and at the same time improves the occupancy grid estimation by a significant margin over the state of the art. Our approach also outperforms the strong baseline of directly back-projecting 2D vision-language features into 3D via LiDAR and does not require LiDAR at test-time. This work opens-up the possibility of large-scale open-vocabulary 3D scene understanding driven by natural language.

Acknowledgments. This work was supported by the European Regional Development Fund under the project IMPACT (no. CZ.02.1.010.00.015_0030000468), by the Ministry of Education, Youth and Sports of the Czech Republic through the e-INFRA CZ (ID:90140), and by CTU Student Grant SGS21184OHK33T37. This research received the support of EXA4MIND, a European Union´s Horizon Europe Research and Innovation programme under grant agreement N° 101092944. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Commission. Neither the European Union nor the granting authority can be held responsible for them.

References

[1] Maxim Berman, Amal Rannen Triki, and Matthew B Blaschko. The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In *CVPR*, 2018. 2

[2] Alexandre Boulch, Corentin Sautier, Björn Michele, Gilles Puy, and Renaud Marlet. ALSO: Automotive lidar self-supervision by occupancy estimation. In *CVPR*, 2022. 3

[3] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *CVPR*, 2020. 3, 4

[4] Anh-Quan Cao and Raoul de Charette. Monoscene: Monocular 3d semantic scene completion. In *CVPR*, 2022. 2

[5] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *CVPR*, 2022. 2

[6] Karan Desai and Justin Johnson. Virtex: Learning visual representations from textual annotations. In *CVPR*, 2021. 2

[7] Albert Gordo and Diane Larlus. Beyond instance-level image retrieval: Leveraging captions to learn a global visual representation for semantic retrieval. In *CVPR*, 2017. 2

[8] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. *arXiv preprint arXiv:2104.13921*, 2021. 3

[9] Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Tri-perspective view for vision-based 3d semantic occupancy prediction. In *CVPR*, 2023. 1, 2, 3, 4

[10] Krishna Murthy Jatavallabhula, Alihusein Kuwajerwala, Qiao Gu, Mohd Omama, Tao Chen, Shuang Li, Ganesh Iyer, Soroush Saryazdi, Nikhil Keetha, Ayush Tewari, Joshua B. Tenenbaum, Celso Miguel de Melo, Madhava Krishna, Liam Paull, Florian Shkurti, and Antonio Torralba. Conceptfusion: Open-set multimodal 3d mapping. In *RSS*, 2023. 2

[11] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021. 2

[12] Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. LERF: Language embedded radiance fields. *arXiv preprint arXiv:2303.09553*, 2023. 2

[13] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 3

[14] Gen Li, Nan Duan, Yuejian Fang, Ming Gong, and Daxin Jiang. Unicoder-VL: A universal encoder for vision and language by cross-modal pre-training. In *AAAI*, 2020. 2

[15] Junnan Li, Ramprasaath R. Selvaraju, Akhilesh Gotmare, Shafiq R. Joty, Caiming Xiong, and Steven Chu-Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. In *NeurIPS*, 2021. 2

[16] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 2

[17] Mert Bülent Sariyildiz, Julien Perez, and Diane Larlus. Learning visual representations with caption annotations. In *ECCV*, 2020. 2

[18] Chong Zhou, Chen Change Loy, and Bo Dai. Extract free dense labels from clip. In *ECCV*, 2022. 1, 2, 3, 4