Unsupervised Semantic Segmentation
Through Depth-Guided Feature Correlation and Sampling

Leon Sick
Ulm University

Dominik Engel
Ulm University

Pedro Hermosilla
TU Vienna

Timo Ropinski
Ulm University

Abstract

Traditionally, training neural networks to perform semantic segmentation requires expensive human-made annotations. But more recently, advances in the field of unsupervised learning have made significant progress on this issue and towards closing the gap to supervised algorithms. To achieve this, semantic knowledge is distilled by learning to correlate randomly sampled features from images across an entire dataset. In this work, we build upon these advances by incorporating information about the structure of the scene into the training process through the use of depth information. We achieve this by (1) learning depth-feature correlation by spatially correlating the feature maps with the depth maps to induce knowledge about the structure of the scene and (2) exploiting farthest-point sampling to more effectively select relevant features by utilizing 3D sampling techniques on depth information of the scene. Finally, we demonstrate the effectiveness of our technical contributions through extensive experimentation and present significant improvements in performance across multiple benchmark datasets.

1. Introduction

Semantic segmentation plays a critical role in many of today's vision systems in a multitude of domains. These include, among others, autonomous driving [15], medical applications [33, 39], and many more [9, 27, 41, 45]. Until recently, the main body of research in this area was focused on supervised models that require a large amount pixel-level annotations for training. Not only is sourcing this image data often a labor intensive process, but also annotating the large datasets required for good performance comes at a high price. Several benchmark datasets report their annotation times. For example, the MS COCO dataset [28] required more than 28K hours of human annotation for around 164K images, and annotating a single image in the Cityscapes dataset [11] took 1.5 hours on average. These costs have triggered the advent of unsupervised semantic segmentation [10, 16, 20, 35], which aims to remove the need for labeled training data in order to train segmentation models. Recently, work by Hamilton *et al.* [16] has accelerated the progress towards removing the need for labels to achieve good results on semantic segmentation tasks. Their model, STEGO, uses a DINO-pretrained [7] Vision Transformer (ViT) [13] to extract features that are then distilled across the entire dataset to learn semantically relevant features, using a contrastive learning approach. The to-be-distilled features are sampled randomly from feature maps produced from the same image, k-NN matched images as well as other negative images. Seong *et al.* [35] build on this process by trying to identify features that are most relevant to the model by discovering hidden positives. Their work exposes an inefficiency of random sampling in STEGO as hidden positives sampling leads to significant improvements. However, both approaches only operate on the pixel space and therefore fail to take into account the spatial layout of the scene. Not only do we humans perceive the world in 3D, but also previous work [5, 8, 18, 37, 38, 44] has shown that supervised semantic segmentation can benefit greatly from spatial information during training. Inspired by these observations, we propose to incorporate spatial information in the form of depth maps into the STEGO training process. Depth is considered a product of vision and does not provide a labeled training signal. To obtain depth information for the benchmark image datasets in our evaluations, we make use of an off-the-shelf zero-shot monocular depth estimator to obtain spatial information of the scene. This allows us to incorporate the depth information without the need for human annotations or sensor ground truth.

With our method, *DepthG*, we propose to **(1)** guide the model to learn a rough spatial layout of the scene, since we hypothesize this will aid the network in differentiating objects much better. We achieve this by extending the contrastive process to the spatial dimension: We do not limit the model to learning only Feature-Feature Correlations, but also *Depth-Feature Correlations*. Through this process, the model is guided towards pulling apart the features with high

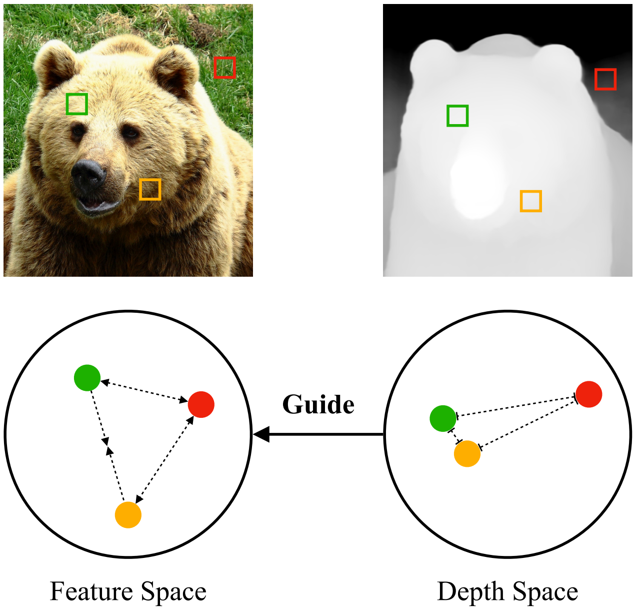


Figure 1. **Guiding the feature space for unsupervised segmentation with depth information.** Our intuition behind the proposed approach is simple: For locations in the 3D space with a low distance, we guide the model to map their features closer together. Vice versa, the features are learned to be drawn apart in feature space if their distance in the metric space is large.

distances in 3D space, as well as mapping them closer together if their distance is low depth space. Figure 1 visualizes this process.

With the information about the spatial layout of the scene present, we furthermore propose to **(2)** spatially inform our feature sampling process by utilizing *Farthest-Point Sampling (FPS)* [14, 31] on the depth map, which equally samples scenes in 3D. We show these techniques in combination make our approach to unsupervised segmentation highly effective, since in our evaluations across multiple benchmark datasets, we demonstrate state-of-the-art performance. We include a short video presenting our method as part of the supplementary materials. To the best of our knowledge, we are the first to propose a mechanism to incorporate 3D knowledge of the scene into unsupervised learning for 2D images *without* encoding depth maps as part of the network input. This design aspect of our method alleviates the risk of the model developing an input dependency. Therefore, our approach does not rely on depth information during inference and is not affected by availability or quality of such depth information.

2. Related Work

2.1. Unsupervised Semantic Segmentation

Recent works [10, 16, 20, 35] have attempted to tackle semantic segmentation without the use of human annotations. Ji *et al.* [20] propose IIC, a method that aims to maximize the mutual information between different augmented versions of an image. PiCIE, published by Cho *et al.* [10], introduces an inductive bias based on the invariance to photometric transformations and equivariance to geometric manipulations. DINO [7] often serves as a critical component to unsupervised segmentation algorithms, since the self-supervised pre-trained ViT can produce semantically relevant features. Recent work by Seitzer *et al.* [34] builds upon this ability by training a model with slot attention [30] to reconstruct the feature maps produced by DINO from the different slots. The features of their object-centric model are clustered with k-means [29] where each slot is associated with a cluster. In their 2021 work, Hamilton *et al.* [16] have also built upon DINO features by introducing a feature distillation process with features from the same image, k-NN retrieved examples as well as random other images from the dataset. Their learned representations are finally clustered and refined with a CRF [22] for semantic segmentation. While STEGO's feature selection process is random, Seong *et al.* [35] introduce a more effective sampling strategy by discovering hidden positives. During training, they form task-agnostic and task-specific feature pools. For an anchor feature, they then compute the maximum similarity to any of the pool features and sample locations in the image with greater similarity than the determined value. A more detailed introduction to STEGO is provided in Section 3.1.

2.2. Depth For Semantic Segmentation

Previous research [5, 17, 18, 38, 40, 43] has sought to incorporate depth for semantic segmentation in different settings. Wang *et al.* [38] propose to use depth for adapting segmentation models to new data domains. Their method adds depth estimation as an auxiliary task to strengthen the prediction of segmentation tasks. Furthermore, they approximate the pixel-wise adaption difficulty from source to target domain through the use of depth decoders. Work by Hoyer *et al.* [18] explores three further strategies of how depth can be useful for segmentation. First, they propose using a shared backbone to share learning features for segmentation and self-supervised depth estimation, similar to Wang *et al.* [38]. Second, they use depth maps to introduce a data augmentation that is informed by the structure of the scene. And lastly, they detail the integration of depth into an active learning loop as part of a student-teacher setup. Another work by Hou *et al.* [17] incorporates depth information into a pre-training algorithm with the aim to learn better representations for semantic segmentation. In their work, Mask3D, they propose to incorporate depth in the form of a 3D prior by formulating a reconstruction task that operates on masked RGB and depth patches, enabling them to learn more useful features for semantic segmentation.

3. Method

In the following, we detail our proposed method for guiding unsupervised segmentation with depth information. An overview of our approach is presented in Figure 2.

3.1. Preliminary

Our approach builds upon work by Hamilton *et al.* [16]. In their work, each image is 5-cropped and k-NN correspondences between these images are calculated using the DINO ViT [7]. Generally, STEGO uses a feature extractor $\mathcal{F}: \mathbb{R}^{3\times H_{\text{in}}\times W_{\text{in}}} \rightarrow \mathbb{R}^{C\times H\times W}$ with input image height H_{in} and width W_{in}, to calculate a feature map $f \in \mathbb{R}^{C\times H\times W}$ with height H, width W and feature dimension C from the input image. These features are then further encoded by a segmentation head $\mathcal{S}: \mathbb{R}^{C\times H\times W} \rightarrow \mathbb{R}^{Q\times H\times W}$ to calculate the code space $s \in \mathbb{R}^{Q\times H\times W}$ with code dimension Q. With the goal of forming compact clusters and amplifying the correlation of the learned features, let $f_i := \mathcal{F}(x_i)$ and $f_j := \mathcal{F}(x_j)$ be feature maps for a given input pair of x_i and x_j, which are then used to calculate $s_i := \mathcal{S}(f_i)$ and $s_j := \mathcal{S}(f_j)$ from the segmentation head $\mathcal{S}$. In practice, STEGO samples N^2 vectors from the feature map during training. Hamilton *et al.* [16] introduce the concept of constructing the feature correspondence tensor $\boldsymbol{F} \in \mathbb{R}^{H\times W\times H\times W}$ as follows:

$$\boldsymbol{F}_{hw,uv} = \frac{f_i^{hw} \cdot f_j^{uv}}{\|f_i^{hw}\| \|f_j^{uv}\|} \tag{1}$$

where $\cdot$ denotes the dot product. Using the same formula we obtain $\boldsymbol{S}$ using s_i, s_j. Consequently, the feature correlation loss is defined as:

$$\mathcal{L}_{\text{Corr}} := -\sum_{hw,uv} (\boldsymbol{F}_{hw,uv} - b) \max(\boldsymbol{S}_{hw,uv}, 0) \tag{2}$$

where b is a scalar bias hyperparameter. Empirical evaluations from STEGO have shown that applying spatial centering to the feature correlation loss along with zero-clamping further improves performance [16]. These correlations are calculated for two crops from the same image ($\mathcal{L}_{\text{self}}$) and one from a different but similar image, determined by the k-NN correspondence pre-processing ($\mathcal{L}_{\text{knn}}$). Finally, negative images are sampled randomly ($\mathcal{L}_{\text{random}}$). The final loss is a weighted sum of the different losses where each of them has their individual weight λ_i:

$$\mathcal{L}_{\text{STEGO}} = \lambda_{\text{self}}\mathcal{L}_{\text{self}} + \lambda_{\text{knn}}\mathcal{L}_{\text{knn}} + \lambda_{\text{random}}\mathcal{L}_{\text{random}} \tag{3}$$

After training, the inferred feature maps for a test image are clustered using k-means and refined with a conditional random field (CRF) [22].

3.2. Depth Map Generation

Since in many cases, depth information about the scene is not readily available, we make use of recent progress in the field of monocular depth estimation [1–3, 25, 32] to obtain depth maps from RGB images. Recently, methods from this field have made significant advances in zero-shot depth estimation, i.e. predicting depth values for scenes from data domains not seen during training. This property makes them especially suitable for our method, since it enables us to obtain high-quality depth predictions for a wide variety of data domains without ever re-training the depth network. This property also limits the computational cost for our method. For our method, we experiment with different state-of-the-art monocular depth estimators, detailed in Section 5, and found ZoeDepth [3] to perform best in our evaluations. Given a cropped RGB image x_i, we use the monocular depth estimator $\mathcal{M}$ together with average pooling to predict depth $d_i \in [0,1]^{H\times W}$ at feature resolution:

$$d_i = \text{pool}(\mathcal{M}(x_i)) \tag{4}$$

The average pooling operation is used to match the dimensions of the feature map, which is required to sample non-overlapping locations at the patch resolution.

3.3. Depth-Feature Correlation Loss

With our *Depth-Feature Correlation* loss, we aim to enforce spatial consistency in the feature map by transferring the distances from the depth map to the latent feature space.

In contrastive learning, the network is incentivized to decrease the distance in feature space for similar instances, therefore learning to map their latent representations closer together. Likewise, different instances are drawn further apart in feature distance. We assume the same concept to be true in 3D space: The spatial distance between two points from the same depth plateau is smaller, while the distance between a point in the foreground and one in the background is larger. Since, in both spaces, the concept of measuring difference is represented by the distance between two points, we propose to align them through our concept of *Depth-Feature Correlation*: For large distances in the 3D space, we encourage the network to produce vectors that are further apart, and vice versa. With this, we induce the model with knowledge about the spatial structure of the scene, enabling it to better differentiate between objects. To achieve this we construct the depth correspondence tensor similar to the feature correspondence from Equation 1. The depth correspondence tensor $\boldsymbol{D}$ is computed from the depths of two different image crops as follows:

$$\boldsymbol{D}_{hw,uv} = d_i^{hw} d_j^{uv}, \tag{5}$$

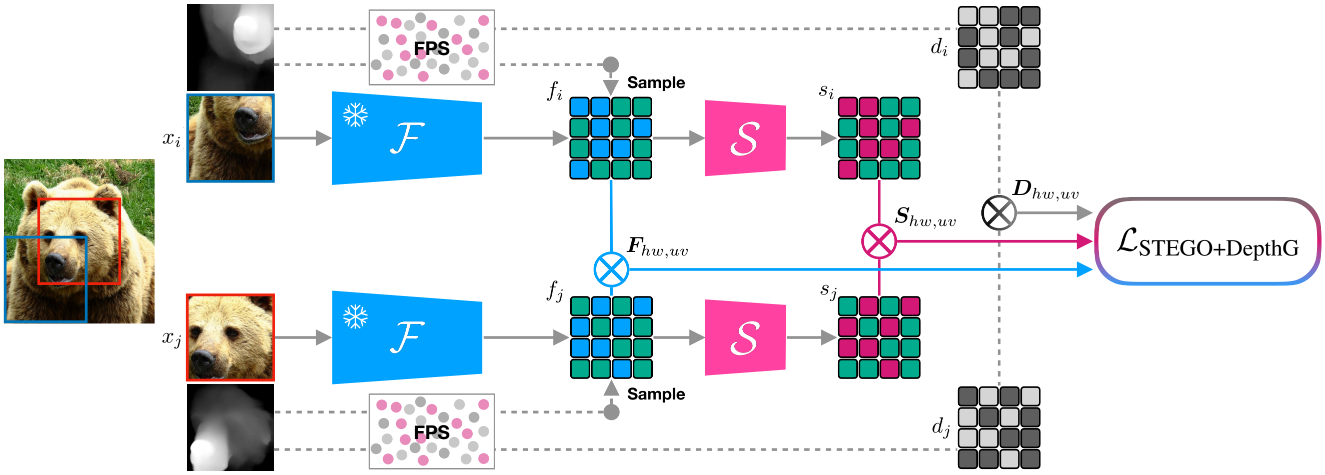


Figure 2. **Overview of the DepthG training process.** After 5-cropping the image, each crop is encoded by the DINO-pretrained ViT $\mathcal{F}$ to output a feature map. Using farthest point sampling (FPS), we sample the 3D space equally and convert the coordinates to select samples in the feature map. The sampled features are further transformed by the segmentation head $\mathcal{S}$. For both feature maps, the correlation tensor is computed. Following, we sample the depth map at the coordinates obtained by FPS and compute a correlation tensor in the same fashion. Finally, we compute our *Depth-Feature Correlation* loss and combine it with the feature distillation loss from STEGO. We guide the model to learn depth-feature correlation for crops of the same image, while the feature distillation loss is also applied to k-NN-selected and random images.

where (h, w) and (u, v) represent the pixel positions in the depth maps d_i and d_j respectively. Together with the zero clamping, our *Depth-Feature Correlation* loss is defined as:

$$\mathcal{L}_{\text{DepthG}} = -\sum_{hw,uv} (\boldsymbol{D}_{hw,uv} - b_{\text{DepthG}}) \max(\boldsymbol{S}_{hw,uv}, 0) \tag{6}$$

where $\boldsymbol{D}_{hw,uv}$ represents the depth correlation tensor, b_{DepthG} is the bias for our loss term, and $\boldsymbol{S}_{hw,uv}$ represents the feature correlation tensor computed from the output features of the segmentation head $\mathcal{S}$. By also using zero-clamping, we limit erroneous learning signals that aim to draw apart instances of the same class if they have large spatial differences. With this, we extend the STEGO loss so it can be formulated as follows:

$$\mathcal{L}_{\text{STEGO+DepthG}} = \mathcal{L}_{\text{STEGO}} + \lambda_{\text{DepthG}} \mathcal{L}_{\text{DepthG}} \tag{7}$$

with *Depth-Feature Correlation* weight λ_{DepthG}. By inducing depth knowledge during training *without* encoding the depth maps as part of the model input, our model can predict spatially informed segmentations on RGB images with its distilled knowledge and does not rely on depth to be available at test time.

3.4. Depth-Guided Feature Sampling

We also aim to make the feature sampling process informed by the spatial layout of the scene. To perform sampling in the 3D space, we transform the downsampled depth map $d(x_i)$ into a point cloud with points $\{p_1, p_2, ..., p_n\}$. On this point cloud, we apply farthest point sampling (FPS) [14], in an iterative fashion by always selecting the next point p_k as the point with the maximum distance in 3D space with respect to the already sampled points $\{p_1, p_2, ..., p_{k-1}\}$. After having sampled N^2 points, we end up with a set of samples $\{p_1, p_2, ..., p_{N^2}\}$ which are consequently converted to 2D sampling indices for the feature maps f and g. In contrast to the data-agnostic random sampling applied in STEGO, our feature selection process takes into account the geometry of the input scene and covers the spatial structure more equally. In our ablations in Section 5, we show this scene coverage from FPS of the depth space further increases the effectiveness of our *Depth Feature Correlation* loss, due to the increased diversity in selected 3D locations. We show a visual comparison between random and farthest point sampling in Figure 5.

3.5. Guidance Scheduling

While our *Depth-Feature Correlation* loss is effective at enriching the model's learning process with spatial information of the scene, we aim to alleviate the danger of it interfering with the learning of feature correlations during model training. We hypothesize that our model most greatly benefits from depth information in the beginning of training when its only knowledge is encoded in the features maps by the frozen ViT backbone. To give it a head start, we increase the weight of our *Depth-Feature Correlation* loss in the beginning and gradually decrease its influence during training. Vice versa, the distillation process in the feature space will increasingly emphasised as the training progresses. In this way, the network builds upon the already learned rough spa-

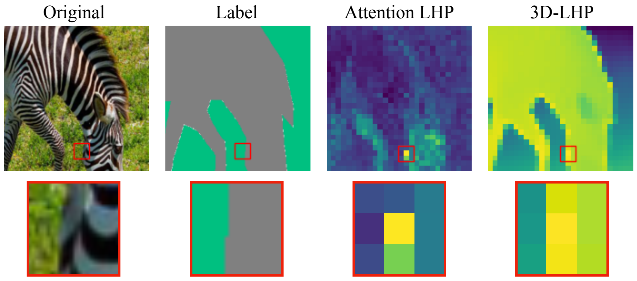


Figure 3. **Local Hidden Positives.** We visualize the use of depth and attention maps for local hidden positives. For this visualization, we sample the respective propagation maps at the yellow patch in the center of the crops. We observe the depth map to have sharper borders and more consistent propagation values. We experiment with both propagation strategies in Section 5.

tial structure of the scene achieved through our depth guidance. Therefore, we implement an exponential decay of the weight λ_{DepthG} and bias b_{DepthG} of our loss component. We ablate on the use of guidance scheduling in the appendix.

3.6. Local Hidden Positives In 3D Space

We further explore the combination of our method with Hidden Positives [35], which also builds upon STEGO. As we demonstrate as part of our ablation on the individual influence of our contributions in Section 5, our feature sampling method, implemented through farthest point sampling, is integral to the effectiveness of *DepthG*. Therefore, we decide not to replace it with the global hidden positives sampling from Seong *et al*. [35]. Instead, we implement a depth-informed variant of local hidden positives, *3D-LHP*, where the loss for an individual patch is propagated to eight neighboring patches proportionally to their attention values obtained from the feature extractor. We modify this strategy by instead propagating the learning signal to the closest patches *in 3D space* proportional to their relative distances, using the point cloud described in Section 3.4. As visualized in Figure 3, we observe that our depth-based propagation map has sharper and more consistent surfaces. To propagate the learning signal to the selected patches, we follow Hidden Positives and mix the features coming from the segmentation head $\mathcal{S}$ in proportion to their propagation values (point distances). The mixed representations are then fed through an additional projection head $\mathcal{P}$. We then calculate $\mathcal{L}_{\text{STEGO+DepthG}}$ again for the produced output features and combine it with the loss from the segmentation head output.

4. Experiments

Datasets and Models. We conduct experiments on the COCO-Stuff [4], Cityscapes [11] and Potsdam-3 datasets. COCO-Stuff contains a wide variety of real-world scenes. In our evaluation, we follow [16, 20, 35] and provide results on the coarse class split, COCO-Stuff 27. In contrast, Cityscapes contains traffic scenes from 50 cities from a driver-like viewpoint. Lastly, the Potsdam-3 dataset is composed of aerial, top-down images of the city of Potsdam. We use the DINO [7] backbones ViT-Small (ViT-S) and ViT-Base (ViT-B), which were pre-trained in a self-supervised manner on ImageNet-1k [12]. We choose the models with a patch size of 8 × 8, since they have been shown to perform best for semantic segmentation due to the higher resolution of the resulting feature maps [16, 35]. In the result tables, we refer to *DepthG* models trained with our *Depth-Feature Correlation* loss and FPS as **Ours**. Models that utilize 3D-LHP propagation with depth maps in addition are displayed as **Ours w/ 3D-LHP**. We compare our approach against competing methods which were never trained with human annotations, i.e. human labels or language supervision, neither in the feature extractor nor the segmentation training.

Evaluation Protocols. Similar to STEGO and related work [16, 35], we evaluate our models in the unsupervised, clustering-based setting, as well as linear probing. Since the output of our model is a pixel-level map of features and not class labels, these features are consequently clustered. Following, the pseudo-labeled clusters are aligned with the ground truth labels through Hungarian matching [23, 24] across the entire validation dataset. To perform linear probing, an additional linear layer is added on top of the model and trained with cross-entropy loss to learn classification of the features.

Setting		Unsupervised		Linear	
Method	Model	Acc.	mIoU	Acc.	mIoU
IIC [20]	R18+FPN	21.8	6.7	44.5	8.4
PiCIE [10]	R18+FPN	48.1	13.8	54.2	13.9
PiCIE+H [10]	R18+FPN	50.0	14.4	54.8	14.8
DINO [7]	ViT-S/8	28.7	11.3	68.6	33.9
ACSeg [26]	ViT-S/8	16.4	-	-	-
TransFGU [42]	ViT-S/8	17.5	52.7	-	-
STEGO + HP [35]	ViT-S/8	**57.2**	24.6	**75.6**	**42.7**
STEGO [16]	ViT-S/8	48.3	24.5	74.4	38.3
+ **Ours**	ViT-S/8	56.3	25.6	73.7	38.9
+ **Ours w/ 3D-LHP**	ViT-S/8	55.1	**26.7**	73.9	37.8
DINO [7, 16]	ViT-B/8	30.5	9.6	66.8	29.4
DINOSAUR [34]*	ViT-B/8	44.9	24.0	-	-
STEGO [16]	ViT-B/8	56.9	28.2	**76.1**	41.0
+ **Ours**	ViT-B/8	**58.6**	**29.0**	75.5	**41.6**

Table 1. **Evaluation on COCO-Stuff 27.** We report results on COCO-Stuff with 27 high-level classes. Overall, our method outperforms STEGO and HP on unsupervised segmentation with the ViT-B/8, while showing competitive performance for the ViT-S/8. *Results obtained without post-processing optimization.

4.1. COCO-Stuff

We present our evaluation on COCO-Stuff 27 in Table 1. For the ViT-S/8, our experiments show that *Ours* is able to improve upon STEGO in most metrics, with improved unsupervised accuracy by **+8.0%** and unsupervised mIoU increased by **+1.1%**. *Ours w/ 3D-LHP* further increases this mIoU delta to **+1.8%**, highlighting the effectiveness of our 3D-information propagation strategy in combination with *DepthG*. When comparing our approach to Hidden Positives, a method with more computational overhead, for the ViT-S/8, we show competitive performance for unsupervised accuracy and outperform their approach by **+1.0%** on unsupervised mIoU with *Ours* and **+1.7%** with *Ours w/ 3D-LHP*. When using the DINO ViT-B/8 encoder, our approach again outperforms STEGO, as well as all other presented methods on unsupervised metrics. Most notably, we are able to increase the unsupervised mIoU by **+0.8%**.

4.2. Cityscapes

We further evaluate our approach on the Cityscapes dataset [11], consisting of various scenes from 50 different cities. We follow the training setting from STEGO and, contrary to all other datasets, do not sample point-wise but use the full feature map for our learning process along with the full depth map. As can be seen in Table 2, our method significantly outperforms STEGO as well as Hidden Positives. For unsupervised mIoU, while Hidden Positives decreased performance compared to STEGO, we observe that our approach to achieves a **+2.1%** increase. Similarly, we report state-of-the-art performance in accuracy, building upon Hidden Positives' already impressive improvements over STEGO and outperforming it by **+2.1%**.

4.3. Potsdam

Our model is further evaluated on the Potsdam-3 dataset, containing aerial images of the German city of Potsdam. Contrary to the other benchmarks, which contain images in a first-person perspective, Potsdam-3 contains only birds-eye-view images, a perspective that is considered out-of-distribution for the monocular depth estimator. Despite this inherent limitation of our approach for aerial data, we are able to demonstrate relatively commendable performance in Table 3 by improving STEGO's performance and reporting state-of-the-art performance for the ViT-S backbone. In contrast, Hidden Positives [35] use a ViT-B/8, and with roughly twice the parameters as ours, reach an accuracy of 82.4%. We present a visual overview of the predicted Potsdam depth maps in the appendix.

Method	Model	U. Acc	U. mIoU
IIC [20]	R18+FPN	47.9	6.4
PiCIE [10]	R18+FPN	65.6	12.3
DINO [7]	ViT-B/8	43.6	11.8
STEGO + HP [35]	ViT-B/8	79.5	18.4
STEGO [16]	ViT-B/8	73.2	21.0
+ **Ours**	ViT-B/8	**81.6**	**23.1**

Table 2. **Results on Cityscapes.** We report unsupervised accuracy and mIoU on Cityscapes. Our method outperforms both STEGO variants by substantial margins. Notably, our method is the first to improve upon unsupervised mIoU.

Method	Model	U. Acc.
CC [19]	VGG11	63.9
DeepCluster [6]	VGG11	41.7
IIC [20]	VGG11	65.1
DINO [7, 21]	ViT-S/8	71.3
STEGO [16, 21]	ViT-S/8	77.0
+ **Ours**	ViT-S/8	**80.4**
+ **Ours w/ 3D-LHP**	ViT-S/8	**80.4**

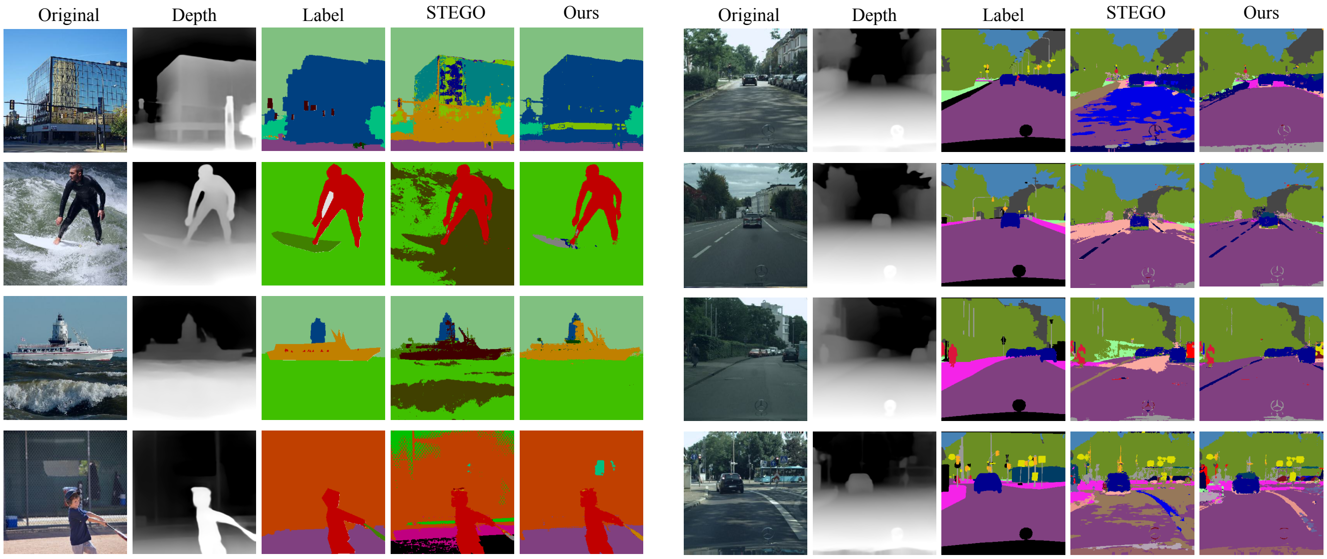
Table 3. **Results on Potsdam.** We report unsupervised accuracy on the Potsdam dataset. Our method is able to improve upon STEGO. We hypothesize that with a zero-shot depth estimator more suitable for aerial images, the results for our method could further improve.

Method	U. mIoU.	U. Acc
STEGO [16]	24.5	48.3
+ Depth-Feature Correlation (1)	24.7	51.2
+ FPS (2)	24.6	49.1
+ 3D-LHP (3)	25.2	48.5
+ **Ours** (1, 2)	25.6	**56.3**
+ **Ours w/ 3D-LHP** (1, 2, 3)	**26.7**	55.1

Table 4. **Effect of our contributions.** We compare our individual contributions and the combination of all contributions.

4.4. Qualitative Results

We present qualitative results of our method in Figure 4 and compare with segmentation maps from STEGO. On multiple occasions, our depth guidance reduces erroneous predictions from the model caused by visual irritations in the pixel space. In the example of the boy with the baseball bat in Figure 4a, false classifications from STEGO are caused by shadows on the ground. Our model is able to correct this. Furthermore, it goes beyond the noisy label and also correctly classifies the glimpse of a plant that can be seen through a hole in the background. This is an indication that our model does not overfit to the depth map, since this visual cue is only observable in the pixel space, but not the depth map.



(a) COCO-Stuff

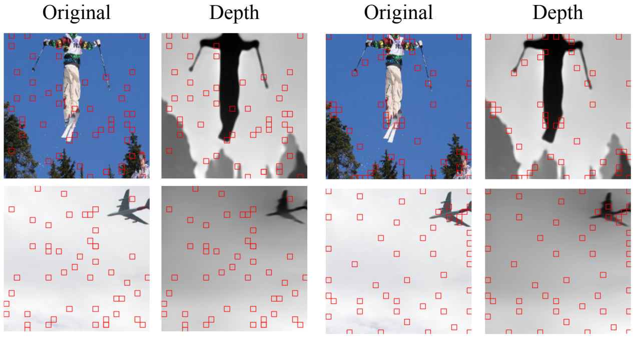
(b) Cityscapes

Figure 4. **Qualitative results.** We show qualitative differences for plain STEGO compared to STEGO with our depth guidance, using ViT-S models for COCO and ViT-B for Cityscapes. Where STEGO struggles to differentiate instances, our model is able to correct this and successfully separates them for segmentation. In the case of the building in (a), our method alleviates visual irritations from the pixel space and corrects the segmentation of the building. In (b), our model is able to better handle visual inconsistencies from shadows.

5. Ablations

Individual Influence. We investigate the effect of our technical contributions on training our model with a ViT-S/8 backbone on COCO-Stuff 27. Our observations in Table 4 show that our *Depth-Feature Correlation* loss itself already improves the performance of STEGO. This improvement is further increased through the use of FPS, which enables us to sample the depth space more meaningfully and therefore encourages more diversity in the depth correlation tensor $\boldsymbol{D}_{hw,uv}$. Intuitively, this sampling diversity significantly amplifies our *Depth-Feature Correlation* for aligning the feature space with the depth space. We provide a visual comparison to random sampling in Figure 5 and additional illustrations in the appendix. FPS retrieves more diverse locations and specifically selects locations with depth discontinuities. This naturally benefits the *Depth-Feature Correlation* to learn sharper edges in this area for the output segmentation. Adding local hidden positives with depth maps further increases the unsupervised mIoU, while slightly lowering the accuracy.

Source Of Depth Maps. We investigate the effect of different monocular depth estimators to generate the depth maps used to train our model. In our experiments, we consider three options: The previously mentioned ZoeDepth [3], Kick Back & Relax (KBR) [36] which uses self-supervision to learn depth from Slow-TV videos, as well as MiDaS [32], the base model to ZoeDepth. Empirically, ZoeDepth produces the most accurate zero-shot monodepth results across indoor and outdoor datasets, followed by MiDaS and KBR [3,36]. We provide a qualitative comparison in the appendix. To evaluate the influence of the depth map quality on the performance of our model, we first generate depth maps for COCO-Stuff 27 using each of the introduced models. We then train our model with a ViT-S/8 backbone and report the results in Table 5. We observe that depth maps from ZoeDepth work best with our method, while the model trained with MiDaS depth has an edge over the KBR counterpart.



(a) Random Sampling

(b) Farthest Point Sampling

Figure 5. **Random vs. Farthest Point Sampling.** We observe that random sampling can miss entire structures like trees in the first top and the plane in the bottom row. In contrast, our method meaningfully samples the depth space and selects locations across the different structures and at depth edges. We show further illustrations of FPS in the appendix.

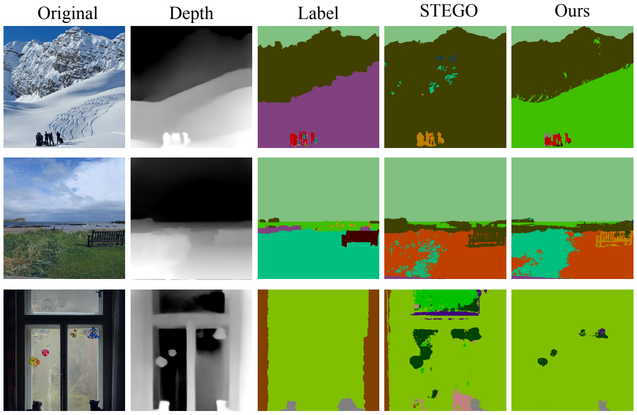


Figure 6. **Failure cases.** We show cases where our model fails to correctly segment and classify the scene. The top row is a prime example where the difference in depth is correctly distilled, though the model fails to correctly classify the snow region.

Signal Propagation With Local Hidden Positives. We ablate the implementation of LHP along with different propagation strategies. As described in Section 3.6, our method takes advantage of the depth information of the scene to propagate the learning signal to patches which are nearby in 3D space. We further implement utilizing the attention map from the DINO backbone and propagate proportionally to their values i.e., the approach utilized in Hidden Positives [35]. While, in Table 6, we show that our approach benefits from signal propagation with 3D-LHP, applying LHP with Attention leads to lower performance. We speculate 3D-LHP is more suitable for our approach since, for a given location, the signal from our *Depth-Feature Correlation* loss is calculated w.r.t. the depth at this sample. Propagating to locations with the same depth does not corrupt this signal, though this can happen with LHP (Attention), since it does not consider depth for propagation.

Computational Cost. Our method only leads to an insignificant increase in runtime versus the baseline STEGO model, since we solely guide the loss as well as the feature sampling and only for *Ours w/ 3D-LHP* add an additional small segmentation head. In contrast, the competing method Hidden Positives [35] relies on a computationally more expensive process to select features and introduces an additional segmentation head to fill their task-specific feature pool. To keep our computational overhead low, we make use of a pre-trained monocular depth estimation network with impressive zero-shot capabilities. We consider task specific training of the depth estimator not a necessity, since the model has zero-shot capabilities that generalize well to different scenes and domains. Therefore, in our experiments on a diverse array of scenes, we do not re-train the depth estimator, and consider the additional computational cost for generating the depth maps to be negligible.

Method	Trained with	U. Acc.	U. mIoU
ZoeDepth [3]	Sensor Depth	**56.3**	**25.6**
MiDaS [32]	Sensor Depth	53.0	25.0
KBR [36]	Self-Supervision	50.6	23.1

Table 5. **Different depth map sources.** We experiment with different monocular depth estimators which were trained with either sensor depth or self-supervision. Overall, the model trained with depth maps from ZoeDepth performs best on COCO-Stuff 27.

LHP	Propagation Strategy	U. Acc.	U. mIoU
✗	-	**56.3**	<u>25.6</u>
✓	3D-LHP (Depth)	<u>55.1</u>	**26.7**
✓	LHP (Attention)	52.6	24.5

Table 6. **LHP Ablation.** We compare the use of depth and attention for local hidden positives. We find that using depth improves unsupervised mIoU, while we find that using attention does not improve our method.

6. Limitations

While we have demonstrated our method's effectiveness for many real-world cases, our method's applicability is limited in settings unsuitable for depth estimation, such as slices of CT scans and other medical data domains. Furthermore, the experiments on Potsdam-3 have shown, our method can improve unsupervised semantic segmentation despite suboptimal viewing perspectives for the monocular depth estimator. We assume the scenario of aerial images represents a rare case where our method would profit from a domain-specific monocular depth estimator. We also present failure cases of our model in Figure 6.

7. Conclusion & Future Work

In this work, we have presented a novel method to induce spatial knowledge of the scene into our model for unsupervised semantic segmentation. We have proposed to correlate the feature space with the depth space and use the 3D information to more meaningfully sample features in a spatially informed way. Furthermore, we have demonstrated that these contributions produce state-of-the-art performance on many real-world datasets and thus foster the progress in unsupervised segmentation. The applicability of our approach for other tasks is further to be explored, since we hypothesize it can be useful beyond unsupervised segmentation as part of other contrastive processes. We consider this to be a promising direction for future work. Furthermore, it remains to be investigated what kind of information could be useful in domains where depth is not an obviously meaningful signal, like medical data.

References

[1] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. Adabins: Depth estimation using adaptive bins. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4009–4018, 2021. 3

[2] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. Localbins: Improving depth estimation by learning local distributions. In *European Conference on Computer Vision*, pages 480–496. Springer, 2022. 3

[3] Shariq Farooq Bhat, Reiner Birkl, Diana Wofk, Peter Wonka, and Matthias Müller. Zoedepth: Zero-shot transfer by combining relative and metric depth. *arXiv preprint arXiv:2302.12288*, 2023. 3, 7, 8

[4] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1209–1218, 2018. 5

[5] Adriano Cardace, Luca De Luigi, Pierluigi Zama Ramirez, Samuele Salti, and Luigi Di Stefano. Plugging self-supervised monocular depth into unsupervised domain adaptation for semantic segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1129–1139, 2022. 1, 2

[6] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European conference on computer vision (ECCV)*, pages 132–149, 2018. 6

[7] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 1, 2, 3, 5, 6

[8] Po-Yi Chen, Alexander H Liu, Yen-Cheng Liu, and Yu-Chiang Frank Wang. Towards scene understanding: Unsupervised monocular depth estimation with semantic-aware representation. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 2624–2632, 2019. 1

[9] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. 2022. 1

[10] Jang Hyun Cho, Utkarsh Mall, Kavita Bala, and Bharath Hariharan. Picie: Unsupervised semantic segmentation using invariance and equivariance in clustering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16794–16804, 2021. 1, 2, 5, 6

[11] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 1, 5, 6

[12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 5

[13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020. 1

[14] Yuval Eldar, Michael Lindenbaum, Moshe Porat, and Yehoshua Y Zeevi. The farthest point strategy for progressive image sampling. *IEEE Transactions on Image Processing*, 6(9):1305–1315, 1997. 2, 4

[15] Di Feng, Christian Haase-Schütz, Lars Rosenbaum, Heinz Hertlein, Claudius Glaeser, Fabian Timm, Werner Wiesbeck, and Klaus Dietmayer. Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges. *IEEE Transactions on Intelligent Transportation Systems*, 22(3):1341–1360, 2020. 1

[16] Mark Hamilton, Zhoutong Zhang, Bharath Hariharan, Noah Snavely, and William T Freeman. Unsupervised semantic segmentation by distilling feature correspondences. In *International Conference on Learning Representations*, 2021. 1, 2, 3, 5, 6

[17] Ji Hou, Xiaoliang Dai, Zijian He, Angela Dai, and Matthias Nießner. Mask3d: Pre-training 2d vision transformers by learning masked 3d priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13510–13519, 2023. 2

[18] Lukas Hoyer, Dengxin Dai, Yuhua Chen, Adrian Koring, Suman Saha, and Luc Van Gool. Three ways to improve semantic segmentation with self-supervised depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11130–11140, 2021. 1, 2

[19] Phillip Isola, Daniel Zoran, Dilip Krishnan, and Edward H Adelson. Learning visual groups from co-occurrences in space and time. *arXiv preprint arXiv:1511.06811*, 2015. 6

[20] Xu Ji, Joao F Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9865–9874, 2019. 1, 2, 5, 6

[21] Alexander Koenig, Maximilian Schambach, and Johannes Otterbach. Uncovering the inner workings of stego for safe unsupervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshop*, pages 3788–3797, 2023. 6

[22] Philipp Krähenbühl and Vladlen Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. *Advances in neural information processing systems*, 24, 2011. 2, 3

[23] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955. 5

[24] Harold W Kuhn. Variants of the hungarian method for assignment problems. *Naval research logistics quarterly*, 3(4):253–258, 1956. 5

[25] Jin Han Lee, Myung-Kyu Han, Dong Wook Ko, and Il Hong Suh. From big to small: Multi-scale local planar

guidance for monocular depth estimation. *arXiv preprint arXiv:1907.10326*, 2019. 3

[26] Kehan Li, Zhennan Wang, Zesen Cheng, Runyi Yu, Yian Zhao, Guoli Song, Chang Liu, Li Yuan, and Jie Chen. Acseg: Adaptive conceptualization for unsupervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7162–7172, 2023. 5

[27] Chen Liang-Chieh, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *International Conference on Learning Representations*, 2015. 1

[28] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 1

[29] Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2):129–137, 1982. 2

[30] Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. *Advances in Neural Information Processing Systems*, 33:11525–11538, 2020. 2

[31] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 2

[32] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE transactions on pattern analysis and machine intelligence*, 44(3):1623–1637, 2020. 3, 7, 8

[33] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015. 1

[34] M Seitzer, M Horn, A Zadaianchuk, D Zietlow, T Xiao, C Simon-Gabriel, T He, Z Zhang, B Schölkopf, Thomas Brox, et al. Bridging the gap to real-world object-centric learning. In *International Conference on Learning Representations (ICLR)*, 2023. 2, 5

[35] Hyun Seok Seong, WonJun Moon, SuBeen Lee, and Jae-Pil Heo. Leveraging hidden positives for unsupervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19540–19549, 2023. 1, 2, 5, 6, 8

[36] Jaime Spencer, Chris Russell, Simon Hadfield, and Richard Bowden. Kick back & relax: Learning to reconstruct the world by watching slowtv. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. 7, 8

[37] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Matthieu Cord, and Patrick Pérez. Dada: Depth-aware domain adaptation in semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7364–7373, 2019. 1

[38] Qin Wang, Dengxin Dai, Lukas Hoyer, Luc Van Gool, and Olga Fink. Domain adaptive semantic segmentation with self-supervised depth estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8515–8525, 2021. 1, 2

[39] Risheng Wang, Tao Lei, Ruixia Cui, Bingtao Zhang, Hongying Meng, and Asoke K Nandi. Medical image segmentation using deep learning: A survey. *IET Image Processing*, 16(5):1243–1267, 2022. 1

[40] Yikai Wang, Xinghao Chen, Lele Cao, Wenbing Huang, Fuchun Sun, and Yunhe Wang. Multimodal token fusion for vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12186–12195, 2022. 2

[41] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in Neural Information Processing Systems*, 34:12077–12090, 2021. 1

[42] Zhaoyuan Yin, Pichao Wang, Fan Wang, Xianzhe Xu, Hanling Zhang, Hao Li, and Rong Jin. Transfgu: a top-down approach to fine-grained unsupervised semantic segmentation. In *European conference on computer vision*, pages 73–89. Springer, 2022. 5

[43] Xiaowen Ying and Mooi Choo Chuah. Uctnet: Uncertainty-aware cross-modal transformer network for indoor rgb-d semantic segmentation. In *European Conference on Computer Vision*, pages 20–37. Springer, 2022. 2

[44] Zhenyu Zhang, Zhen Cui, Chunyan Xu, Yan Yan, Nicu Sebe, and Jian Yang. Pattern-affinitive propagation across depth, surface normal and semantic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4106–4115, 2019. 1

[45] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6881–6890, 2021. 1