SemiVL: Semi-Supervised Semantic Segmentation with Vision-Language Guidance

Lukas Hoyer [1,2*]   David Joseph Tan [2]   Muhammad Ferjad Naeem [1]
Luc Van Gool [1,4,5]   Federico Tombari [2,3]
[1] ETH Zurich   [2] Google   [3] TU Munich   [4] KU Leuven   [5] INSAIT Sofia

{lhoyer,mnaeem,vangool}@vision.ee.ethz.ch, {djtan,tombari}@google.de

Abstract

In semi-supervised semantic segmentation, a model is trained with a limited number of labeled images along with a large corpus of unlabeled images to reduce the high annotation effort. While previous methods are able to learn good segmentation boundaries, they are prone to confuse classes with similar visual appearance due to the limited supervision. On the other hand, vision-language models (VLMs) are able to learn diverse semantic knowledge from image-caption datasets but produce noisy segmentation due to the image-level training. In SemiVL, we propose to integrate rich priors from VLM pre-training into semi-supervised semantic segmentation to learn better semantic decision boundaries. To adapt the VLM from global to local reasoning, we introduce a spatial fine-tuning strategy for label-efficient learning. Further, we design a language-guided decoder to jointly reason over vision and language. Finally, we propose to handle inherent ambiguities in class labels by providing the model with language guidance in the form of class definitions. We evaluate SemiVL on 4 semantic segmentation datasets, where it significantly outperforms previous semi-supervised methods. For instance, SemiVL improves the state-of-the-art by +13.5 mIoU on COCO with 232 annotated images and by +6.1 mIoU on Pascal VOC with 92 labels. Project page: github.com/google-research/semivl

1. Introduction

Semantic segmentation models predict pixel-level dense semantic labels for an image. They have important applications in many areas such as autonomous driving, augmented reality, robotics, medical imaging, and remote sensing. However, training such models requires very costly pixel-wise human annotations over a large dataset. To reduce the dependence on large labeled datasets, semi-supervised semantic segmentation aims to effectively learn from a small portion

---

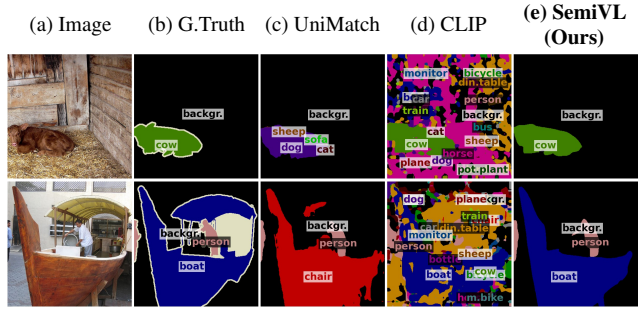*This work was conducted during an internship at Google.

Figure 1. While previous SOTA methods such as UniMatch [68] achieve good segmentation even with only 92 labeled images, they struggle to distinguish classes with similar appearance. In contrast, vision-language training such as CLIP [51] learns rich semantic representations but suffers from noisy segmentation (obtained with [73]). Our SemiVL supplements semi-supervised training with vision-language guidance, combining good segmentation with a rich semantic understanding.

of labeled images while additionally leveraging a large set of unlabeled images. Typical strategies include adversarial training [44, 56] and self-training [55, 67, 68].

While state-of-the-art methods for semi-supervised semantic segmentation such as UniMatch [68] are able to learn good segmentation masks, segments with similar visual features are prone to misclassification, due to the lack of sufficient labeled examples for learning the exact semantic decision boundaries for these classes. This is particularly problematic for rare instances such as a calve or a boat on land as shown in Fig. 1c. Here, the cow (top, green) is confused with a dog (purple) and the boat (bottom, blue) with a chair (red) even though the segmentation is mostly correct.

To better capture semantics, we propose to supplement semi-supervised semantic segmentation with guidance from Vision Language Models (VLM). VLMs such as CLIP [51] are trained on a web-scale image-caption dataset. The diversity of the data and the natural language captions (instead of a fixed set of classes) enables VLMs to capture richer seman-

tic representations. For example, they can correctly identify the cow and the boat in Fig. 1d. However, as they are trained on image level, their features do not localize well and their dense predictions [72] are noisy as shown in Fig. 1d.

In this work, we study how to combine the good localization of semi-supervised training and the rich semantic understanding of VLMs. Based on the findings, we propose SemiVL, which combines both strengths to achieve a good segmentation quality and a fine-grained semantic discriminability. For example, SemiVL correctly segments and classifies the cow and boat in Fig. 1e. To the best of our knowledge, SemiVL is the first work to utilize vision-language guidance for semi-supervised semantic segmentation to mitigate the issue of limited dense labels. Previous works in VLM semantic segmentation (see Sec. 2.2) either operate without dense labels [5, 63, 73], which significantly limits their performance, or use large-scale annotated segmentation datasets [12, 65, 76], which are expensive to obtain. Instead, SemiVL can learn high-quality semantic segmentation in a semi-supervised setting with only a few labels.

Building on consistency regularization [68] for semi-supervised training, SemiVL introduces five components to transfer the power of vision-language modeling to semantic segmentation under the constraint of limited segmentation annotations, which are highlighted in green color in Fig. 3:

(1) Vision-Language Pre-Training (Sec. 3.2): To utilize the rich semantic prior of VLMs, we initialize the semi-supervised training with vision-language pre-training. Compared to the previously used ImageNet pre-training, the VL pre-training on web-scale image-caption pairs provides more diversity and is not limited to a fixed set of classes.

(2) Spatial Fine-Tuning (Sec. 3.3): As the VL pre-training is conducted on image level, the vision model needs to be fine-tuned to achieve good feature localization for semantic segmentation. Due to the limited annotations, the fine-tuning is prone to overfitting and forgetting the rich semantics from the pre-training. Therefore, we introduce parameter-efficient spatial fine-tuning. It only fine-tunes network layers that model interactions between different pixels to improve the localization of features while it freezes layers that operate locally to preserve their semantic reasoning capabilities.

(3) Language-Guided Decoder (Sec. 3.4): To exploit the alignment of vision and language embeddings from VL pre-training, we integrate it into the segmentation decoder. It processes VL similarity maps using decoupled spatial and semantic reasoning. By sharing parameters across classes or pixels, the limited labels are used effectively.

(4) Dense CLIP Guidance (Sec. 3.5): When fine-tuning the VLM towards segmentation, some of the original prior is corrupted and the network drifts to wrong predictions on the unlabeled images. To anchor the semi-supervised training on unlabeled images, we regularize it with the predictions from a *frozen* VLM, which cannot drift. As these predictions
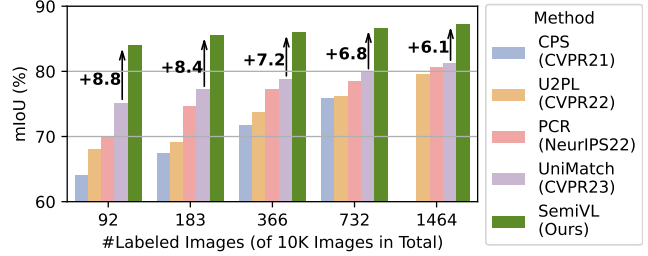

Figure 2. Semi-supervised semantic segmentation on Pascal VOC. Utilizing vision-language guidance, our SemiVL achieves major gains over previous methods, particularly for fewer labels.

are noisy, we only use the ones with a high certainty.

(5) Class Definitions (Sec. 3.6): Depending on the use case, certain concepts can fall into different classes. For example, a person pushing a bicycle could belong to the class pedestrian or rider. However, a small number of annotations might not be sufficient to learn all relevant dataset-specific decision boundaries between classes. Therefore, we utilize the novel capability of SemiVL to provide language guidance in the form of class definitions to the model. These are often part of the annotation guidelines of datasets or can be created with much less effort than mining images of relevant corner cases and labeling them with semantic segmentation.

We evaluate SemiVL on 4 common semantic segmentation datasets, where it significantly outperforms previous semi-supervised methods. For instance, it improves the state of the art by +13.5 mIoU on COCO with 232 annotated images and by +9.7 mIoU on ADE20K with 158 labels. In particular, SemiVL can maintain the performance of previous methods using *only a quarter of the segmentation annotations*. The significance of the improvements is visualized in Fig. 2, showing that SemiVL achieves major gains over previous works for all labeled subsets. In particular, SemiVL improves the performance for few labels, showing its effectiveness for label-efficient learning. In a comprehensive ablation study on Pascal VOC, we show that all 5 components of SemiVL significantly improve the semi-supervised performance. The gains are particularly large when only a few labels are available, demonstrating SemiVL's effectiveness to compensate for limited annotations.

2. Related Work

2.1. Semi-Supervised Semantic Segmentation

Early works [44, 56] in semi-supervised segmentation use unlabeled images in a GAN framework [18] to match their predictions with the distribution of manual labels. Self-training [19, 35] methods generate pseudo-labels for unlabeled images and use them for iterative re-training. To mitigate pseudo-label drift and self-confirmation bias [1], strategies such as confidence-weighting [15, 67], curric-

ula [43, 66, 69], class balancing [21, 22, 25, 31], auxiliary self-supervised tasks [24, 29], contrastive learning [60], symbolic reasoning [38], or soft pseudo-labels [43] can be used. Another popular strategy is consistency regularization [2, 53], which makes predictions invariant to perturbations such as data augmentation [55, 67, 68], mixed samples [16, 31, 49], overlapping crops with different context [26, 27, 34], different model initialization [7], masked images [28], or feature perturbations [68]. Due to its success and simplicity, we base SemiVL on consistency training.

2.2. Semantic Segmentation with VLMs

Vision Language Models (VLMs) such as CLIP and its variants [46–48, 72, 74, 74] utilize a web-scale image-caption dataset to learn a joint embedding space between image and text. Due to the semantically rich and diverse dataset, these models possess great generalization to a wide range of vision tasks[8, 9, 20, 33, 42, 59]. With prompt engineering, VLMs can be improved by ensembling prompts [51] or generating class descriptions by LLMs [47, 50]. Recently, VLMs also gained attention in semantic segmentation. For instance, several zero-shot methods [5, 54, 63, 73] aim to learn segmentation only from image-caption pairs without any dense annotations. MaskCLIP [73] shows that (noisy) segmentation emerges in the CLIP vision encoder. Further techniques include hierarchical grouping [63], retrieval and co-segmentation [54], or text-grounded masking [5]. However, the produced segmentations are noisy due to the lack of dense supervision. In contrast, open-vocabulary segmentation methods achieve better segmentation as they use a large labeled semantic segmentation dataset. Here, CLIP is often used to deal with unknown classes during test time. OpenSeg [17] pools visual features with learned mask proposals. LSeg [36] learns dense visual embeddings that align with CLIP text embeddings. ZegFormer [12], ZSseg [64], and OVSeg [39] predict class-agnostic segmentation masks and classify their crops with a frozen CLIP. ZegCLIP [76] simplifies this into a one-stage approach, learning an attention decoder between CLIP text embeddings and dense vision embeddings. CAT-Seg [9] learns to aggregate CLIP cost maps. SAN [65] trains a side network to adapt a frozen CLIP to segmentation. To avoid overfitting to the training classes, several methods use parameter-efficient training strategies such as prompt tuning [3, 32, 75, 76], partial fine-tuning [9, 30], or adapters [23, 57, 65]. Due to its simplicity, SemiVL also follows a one-stage framework [9, 65, 76]. In contrast to these methods, we do not assume access to a large-scale segmentation dataset.

3. Methods

In semi-supervised semantic segmentation, the training dataset consists of a set of labeled images $\mathcal{D}^l = \{(x_i^l, y_i^l)\}$ and another set of unlabeled images $\mathcal{D}^u = \{x_i^u\}$. In the fol-
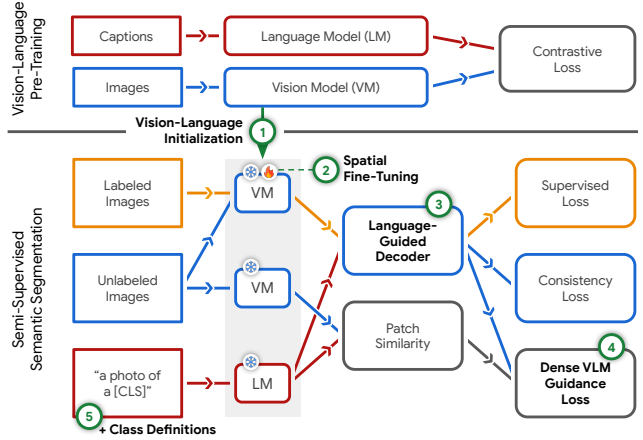

Figure 3. Overview of our **SemiVL framework**. Utilizing the rich semantic representations from vision-language (VL) pre-training (top), we propose 5 strategies (highlighted in green) to guide semi-supervised semantic segmentation (bottom): We use the rich VL prior as initialization (1) and regularization on unlabeled images (4). To adapt VL from image-level to dense reasoning, we introduce a label-efficient fine-tuning strategy (2) and a decoder architecture jointly reasoning about VL (3). Finally, we propose to steer the model with text instructions in the form of class definitions (5).

lowing, we present our SemiVL framework (see Fig. 3). It is based on the popular consistency training approach to utilize unlabeled images during the semi-supervised training (Sec. 3.1). While this already achieves good segmentation quality, it still struggles to obtain precise semantic decision boundaries from the limited supervision. To address this, we propose five components to guide the semi-supervised training with vision-language guidance, which are highlighted in green color in Fig. 3. First, we initialize the semi-supervised training with a pre-trained VLM as it contains rich semantics learned from web-scale image-text data (Sec. 3.2). Second, we only fine-tune the attention layers of the pre-trained vision model to adapt them from image-level to dense reasoning while we freeze the local MLP layers to preserve their semantic reasoning from the pre-training (Sec. 3.3). Third, we introduce language-based reasoning in the decoder to further exploit the vision-language alignment from the pre-training (Sec. 3.4). Fourth, we regularize the training on unlabeled images with predictions from a frozen VLM to anchor the self-training and avoid drift (Sec. 3.5). And fifth, we provide the model with language-based class definitions (Sec. 3.6).

3.1. Consistency Training

While semantic segmentation models are usually trained with a supervised loss $\mathcal{L}_s$ such as the pixel-wise cross-entropy, this is only possible on the labeled images. In order to additionally utilize the unlabeled images for the semi-supervised training, we resort to consistency train-

ing [7, 55, 68]. It is based on the idea that the predictions of the same image should be invariant to different data augmentations or model perturbations. Specifically, the consistency loss term $\mathcal{C}$ drives the model to produce the same predictions p^p under perturbations as the predictions p^u without perturbations, also called pseudo-labels:

$$\mathcal{C}(p^p, p^u) = \sum_{i,j} \mathbb{1}[\max(p_{ij}^u) \geq \tau] H(p_{ij}^p, p_{ij}^u), \quad (1)$$

where i, j are the pixel indices, H is the cross-entropy between p^u and p^p, and τ is a fixed confidence threshold to exclude noisy pseudo-labels p^u from the consistency training. Consistency training is a variant of self-training as the pseudo-labels p^u are generated by the same network as p^p. The pseudo-labels p^u of model f are obtained from the unlabeled images: $p^u = f(x^u)$. The perturbed predictions p^p are obtained under strong data augmentations $\mathcal{A}$: $p^p = f(\mathcal{A}(x^u))$. Additionally, perturbations can also achieved by perturbing the features of the model $p^{fp} = h(\mathcal{P}(g(x^w)))$, where g is the encoder and h the decoder of the model f.

In particular, we follow the state-of-the-art approach Uni-Match [68] and enforce the consistency over two strong data augmentations (p^{p_1}, p^{p_2}) and feature perturbation (p^{fp})

$$\mathcal{L}_u = \frac{1}{2}\mathcal{C}(p^{fp}, p^u) + \frac{1}{4}\mathcal{C}(p^{p_1}, p^u) + \frac{1}{4}\mathcal{C}(p^{p_2}, p^u). \quad (2)$$

The overall semi-supervised loss is $\mathcal{L} = \frac{1}{2}(\mathcal{L}_s + \mathcal{L}_u)$.

3.2. Vision-Language Pre-Training

While consistency training achieves good segmentation as shown by previous works, it is prone to confusion of visually similar classes as can be seen in Fig. 1. This is caused by the model only having a few labeled images to learn the semantic classes. VLMs like CLIP [51] are trained on web-scale image-text datasets that cover almost all semantic classes a vision agent can ever come across. CLIP uses a vision encoder $\mathcal{V}$ to produce embeddings for images and a text encoder $\mathcal{T}$ to embed captions. Both are trained jointly to map to an aligned embedding space, where corresponding images and captions have a high similarity. It is trained in a contrastive manner with ground truth image-caption pairs as positive and shuffled image-caption pairs as negatives. In that way, CLIP learns a representation with a rich semantic understanding of images. CLIP can perform zero-shot image classification by computing the similarities of an image embedding $\mathcal{V}(x)$ with the text embeddings $\mathcal{T}(t_n)$ of multiple class prompts t_n in the form of "a photo of a [CLS$_n$]".

To address the limitations of consistency training, we propose to initialize the semantic segmentation encoder g with the pre-trained VLM vision encoder $\mathcal{V}$ to utilize its rich semantic prior. Our CLIP initialization replaces the prevailing

ImageNet initialization. Compared to image classification pre-training on ImageNet, the VLM pre-training does not require a manually annotated dataset but can be trained on web-crawled image-caption pairs. Further, it can learn richer semantic representations due to the versatile captions, which are not restricted to a specific set of classes.

3.3. Spatial Fine-Tuning

While the vision-language pre-training of CLIP has learned to distinguish fine-grained semantic concepts, it was only trained to optimize image-level features. Therefore, the semantic features are not necessarily spatially aligned with the image content, resulting in noisy segmentations as shown in Fig. 1d. To mitigate this issue, it is necessary to adapt the backbone towards semantic segmentation.

However, due to the limited number of annotated images, the fine-tuning is prone to overfitting and forgetting the rich semantics from the vision-language pre-training. This process can be further reinforced by the self-confirmation bias of the self-training. Inspired by parameter-efficient fine-tuning [9, 30, 32, 57], we introduce spatial fine-tuning to semi-supervised semantic segmentation. A ViT [13] block consists of a multi-head attention layer and a subsequent MLP. Only the attention layer models interactions between different patches, while the MLP operates locally on each patch (similar to a 1×1 convolution). Spatial fine-tuning only fine-tunes the attention layers, which are responsible for spatial reasoning. In that way, the alignment of semantic features and their corresponding image content can be refined for dense predictions. On the other side, the MLP layers are frozen as they do not perform spatial reasoning to preserve the semantic reasoning capabilities of CLIP pre-training. We further fine-tune the position embeddings to attribute for the shift from global to dense reasoning as well as input size change from pre-training to fine-tuning.

3.4. Language-Guided Decoder

A major advantage of the vision-language pre-training is the alignment of the vision and language embeddings, which enables the reasoning about both modalities and their semantic relations. To utilize this capability in the model architecture, we integrate the vision-language alignment in a language-guided decoder architecture, which is visualized in Fig. 4.

For that purpose, we obtain a dense vision-language similarity map $S \in \mathbb{R}^{H \times W \times N}$ with height H, width W and number of classes N as proposed in [73] by computing the patch-wise cosine similarities of the vision embeddings $g(x)$ and the text embeddings $\mathcal{T}(t_n)$ over all classes $n \in [0 .. N]$ with their text prompt t_n

$$S_{ijn} = \frac{g(x)_{ij} \cdot \mathcal{T}(t_n)}{\|g(x)_{ij}\|\|\mathcal{T}(t_n)\|}. \quad (3)$$

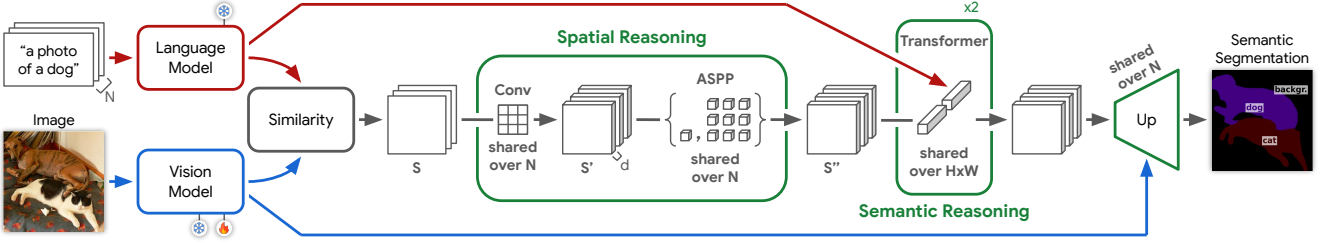Here, i, j denote the patch indices. The embeddings of g and

Figure 4. Overview of our **vision-language-guided architecture**. It is based on dense similarity maps of the vision and text embeddings. These are processed by spatial and semantic reasoning modules and subsequently upsampled. More details are provided in Sec. 3.4.

$\mathcal{T}$ are aligned as g is initialized from $\mathcal{V}$ (see Sec. 3.2). While the original localization of semantics in the similarity map is noisy at the beginning of the training, it is improved during the spatial fine-tuning. The similarity maps are further refined by spatial and semantic reasoning modules.

The spatial reasoning module operates on each class similarity map independently (i.e. $S_n \in \mathbb{R}^{H \times W \times 1}$) and models no inter-class relations. Thereby, the learned spatial reasoning is shared across classes. First, each S_n is processed by a 7×7 convolution to learn local spatial structures and embed them to similarity volumes $S'_n \in \mathbb{R}^{H \times W \times d}$ of d dimensions. Subsequently, a residual ASPP [6] processes the obtained similarity volumes to model long-range context relations, resulting in a combined similarity volume for all classes $S'' \in \mathbb{R}^{H \times W \times N \times d}$.

The semantic reasoning module models the relationship between classes. Each pixel in the similarity volumes $S''_{ij} \in \mathbb{R}^{1 \times 1 \times N \times d}$ is processed independently (no spatial interaction) to share the learned relations between spatial locations. It is implemented as two Transformer [58] blocks. Further, the features are supplemented with the N text embeddings of the class names, which act as semantic anchors.

By decoupling spatial and semantic reasoning, the learned weights can be shared over different classes for spatial reasoning and shared over different locations for semantic reasoning. In that way, the limited annotations can be utilized more effectively and overfitting is reduced.

The common vision transformers operate on a 16 times smaller feature resolution than the input. This limits precise segmentation boundaries. Therefore, we add 2 upsampling blocks [52]. They learn the upsampling using a transpose convolution. Further, skip connections from earlier encoder layers are used. The upsampled features and the skip features are concatenated and fused by two convolutions. For label-efficient learning, the upsampling blocks operate on each class independently. A final convolution maps the d dimensions to one channel to obtain the logits for a class.

3.5. Dense CLIP Guidance

While the previous strategies are designed to best transfer the knowledge from vision-language pre-training to seman-

tic segmentation, the self-training on the unlabeled images can cause a drift of the training to erroneous predictions. Incorrect predictions in p^u are used for the self-training and result in a self-confirmation bias. To anchor the self-training on the unlabeled images and reduce this issue, we guide the consistency training with predictions from a frozen auxiliary CLIP model, which cannot drift.

For this purpose, we extract dense vision-language similarity maps S (see Eq. 3) with a frozen $\mathcal{V}$ from the unlabeled images x_u. The similarity map is processed by a softmax and can be used as a semantic segmentation p^{DC} pseudo-label. As p^{DC} tends to be noisy (see Fig. 3), we only utilize its high confidence predictions exceeding a threshold ζ. To guide the consistency training on the unlabeled images with dense CLIP guidance, we supplement the consistency loss terms $\mathcal{C}$ in Eq. 2 by dense CLIP guidance:

$$
\begin{aligned}
\mathcal{C}(p^p, p^u, p^{DC}) = \mathcal{C}(p^p, p^u) + \\
\lambda_{DC} \sum \mathbb{1}[\max(p^{DC}) \geq \zeta] H(p^p, p^{DC}),
\end{aligned}
\tag{4}
$$

where λ_{DC} weighs the dense CLIP guidance loss term. As the guidance is picked up by the pseudo-labels p^u and spatially refined over the course of the self-training, the guidance is more important at the beginning of the training. Therefore, λ^{DC} is linearly decayed during the training.

3.6. Class Definition Guidance

When the number of labeled samples in the semi-supervised setting is small, they might not be sufficient to capture all relevant decision boundaries between classes for the given dataset. This is particularly problematic for edge cases that are not part of the labeled subset or subcategories that could belong to different classes: Does an armchair belong to the *chair* or *sofa* class? Does a bench belong to the *chair* or *background* class? As similar ambiguities happen when humans annotate a dataset, there are usually annotation guidelines with class definitions for a dataset. For example, Pascal VOC defines: "*chair* includes armchairs, deckchairs but not stools or benches". As SemiVL can process arbitrary text, we propose to provide these annotation guidelines to the model to
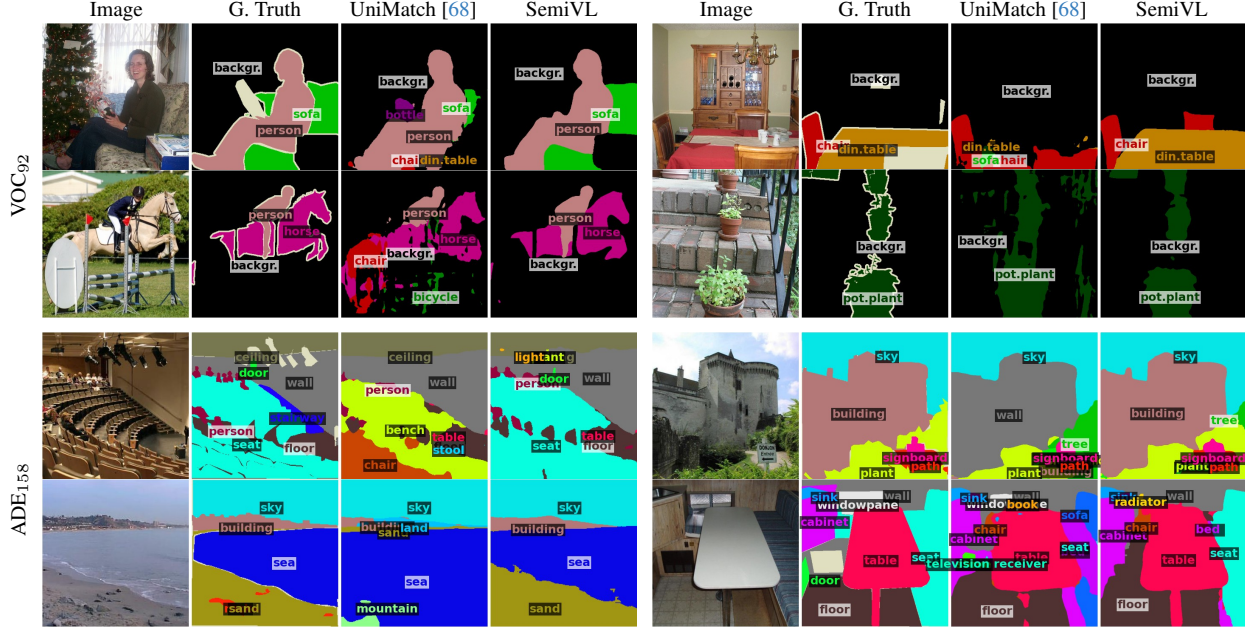
5

Figure 5. **Example predictions** on VOC (92 labels) and ADE20K (158 labels) showing the improved semantic understanding of SemiVL.

better capture the decision boundaries for a dataset. Importantly, providing the class descriptions is much less effort than mining images of all relevant corner cases and labeling them with semantic segmentation. Compared to generic class descriptions from LLMs [47, 50] or synonyms [40], this can better capture dataset-specific decision boundaries, e.g. a stool belongs to the *background* class in Pascal VOC.

Even though it would be possible to provide the text encoder $\mathcal{T}$ directly with the class definitions, CLIP was trained with image captions that do not match such definitions. In particular, they do not handle negations well. Therefore, we build a set of concepts b_a for each class a that contain additional subcategories or descriptions extracted from the class definitions. For example, the concept *armchair* belongs to the class *chair* (i.e. *armchair* $\in b_{chair}$) while *stool* belongs to *background* according to the VOC definition.

To utilize the concepts, we integrate them into the CLIP guidance training. With the frozen CLIP, we infer the vision-text similarity $p^{concept}$ over all concepts $\cup_a b_a$. We aggregate the concepts back to classes using the concept-class assignment from the class definitions. The concept with the highest score determines the predicted class at position ij

$$p_{ija}^{DC} = \max_{b \in b_a} p_{ijb}^{concept} . \tag{5}$$

4. Experiments

4.1. Implementation Details

Architecture: SemiVL uses a ViT-B/16 [13] vision encoder and a Transformer [58] text encoder, both pre-trained

by CLIP [51]. The dense vision embeddings are extracted following [73]. The language-guided decoder uses an initial 7×7 convolution to map the similarity maps to $d{=}128$ channels, a residual ASPP [6] with dilation rates $\{6,12,18\}$ for spatial reasoning, and 2 Transformer blocks [58] with 4 heads for semantic reasoning. For compute and memory efficiency, the semantic reasoning operates on 4×4 average-pooled features maps. The 2 upsampling blocks use a 2×2 transpose convolution, skip connections with 16/32 channels from the first/fourth ViT blocks, and 2 convolutions with GroupNorm [61] to fuse both into 64/32 channels. For Cityscapes, we use the feature map of the first ResNet block as skip connection to handle particularly small segments.

Training: SemiVL is benchmarked on Pascal VOC 2012 [14], COCO [4], ADE20K [71], and Cityscapes [11] using the same labeled subsets as common protocol [7, 60, 68, 77]. It is trained with a batch of 8 labeled and 8 unlabeled images using AdamW [41] for 80/10/40/240 epochs on VOC/COCO/ADE20K/Cityscapes. The inital learning rate is $1\times10^{-4}/4\times10^{-4}/4\times10^{-4}/5\times10^{-5}$ with a 0.9 polynomial decay. Spatial fine-tuning uses a learning rate multiplier of 0.01/0.001/0.001/0.1. We use 801×801 random crops for Cityscapes following [68] and 512×512 for the others. For inference, sliding window evaluation is used. Following UniMatch [68], we set $\tau{=}0.95$, use 50% channel dropout for $\mathcal{P}$, and use random color jitter, grayscale, CutMix, scale, and crop for $\mathcal{A}$. For SemiVL, we set $\zeta{=}0.9$ and use a linear schedule from 0.1 to 0 for λ_{DC}. The class definitions are provided in the supplement. The training is conducted on 4 (for VOC) or 8 (for others) A100 GPUs.

6

Table 1. State-of-the-art comparison on **Pascal VOC**. The mIoU (%) is compared across different splits for the labeled subset $\mathcal{D}^l$. [†] denotes re-produced results in the same setting as SemiVL.

Method		Net	1/115 (92)	1/58 (183)	1/29 (366)	1/14 (732)	1/7 (1464)
PseudoSeg [77]	[ICLR'21]	R101	57.6	65.5	69.1	72.4	–
CPS [7]	[CVPR'21]	R101	64.1	67.4	71.7	75.9	–
ST++ [67]	[CVPR'22]	R101	65.2	71.0	74.6	77.3	79.1
U²PL [60]	[CVPR'22]	R101	68.0	69.2	73.7	76.2	79.5
PCR [62]	[NeurIPS'22]	R101	70.1	74.7	77.2	78.5	80.7
ESL [43]	[ICCV'23]	R101	71.0	74.0	78.1	79.5	81.8
LogicDiag [38]	[ICCV'23]	R101	73.3	76.7	77.9	79.4	–
UniMatch [68]	[CVPR'23]	R101	75.2	77.2	78.8	79.9	81.2
3-CPS [37]	[ICCV'23]	R101	75.7	77.7	80.1	80.9	82.0
ZegCLIP† [76]	[CVPR'23]	ViT-B/16	69.3	74.2	78.7	81.0	82.0
ZegCLIP+UniMatch†	—	ViT-B/16	78.0	80.3	80.9	82.8	83.6
UniMatch† [68]	[CVPR'23]	ViT-B/16	77.9	80.1	82.0	83.3	84.0
Ours	—	ViT-B/16	**84.0**	**85.6**	**86.0**	**86.7**	**87.3**
			(+6.1)	(+5.5)	(+4.0)	(+3.4)	(+3.3)

Table 2. **Class-Wise IoU** on Pascal VOC with 92 Labels.

	Airplane	Backgr.	Bicycle	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Din. Table	Dog	Horse	M.Bike	Person	Plant	Sheep	Sofa	Train	Monitor
UniMatch(ViT)	93	94	69	94	78	76	91	88	94	19	96	56	90	93	82	88	61	94	21	88	71
SemiVL	89	96	78	95	84	84	95	85	97	37	97	72	93	95	90	92	73	94	59	91	68

Table 3. Comparison with state-of-the-art methods on **COCO**.

Method		Net	1/512 (232)	1/256 (463)	1/128 (925)	1/64 (1849)	1/32 (3697)
PseudoSeg [77]	[ICLR'21]	XC65	29.8	37.1	39.1	41.8	43.6
PC²Seg [70]	[ICCV'21]	XC65	29.9	37.5	40.1	43.7	46.1
UniMatch [68]	[CVPR'23]	XC65	31.9	38.9	44.4	48.2	49.8
LogicDiag [38]	[ICCV'23]	XC65	33.1	40.3	45.4	48.8	50.5
UniMatch† [68]	[CVPR'23]	ViT-B/16	36.6	44.1	49.1	53.5	55.0
Ours	—	ViT-B/16	**50.1**	**52.8**	**53.6**	**55.4**	**56.5**
			(+13.5)	(+8.7)	(+4.5)	(+1.9)	(+1.5)

Table 4. Comparison with state-of-the-art methods on **ADE20K**.

Method		Net	1/128 (158)	1/64 (316)	1/32 (632)	1/16 (1263)	1/8 (2526)
CutMix [16]	[BMVC'20]	R101	–	–	26.2	29.8	35.6
AEL [31]	[NeurIPS'21]	R101	–	–	28.4	33.2	38.0
UniMatch [68]	[CVPR'23]	R101	15.6	21.6	28.1	31.5	34.6
UniMatch† [68]	[CVPR'23]	ViT-B/16	18.4	25.3	31.2	34.4	38.0
Ours	—	ViT-B/16	**28.1**	**33.7**	**35.1**	**37.2**	**39.4**
			(+9.7)	(+8.4)	(+3.9)	(+2.8)	(+1.4)

Table 5. Comparison with state-of-the-art methods on **Cityscapes**.

Method		Net	1/30 (100)	1/16 (186)	1/8 (372)	1/4 (744)	1/2 (1488)
PseudoSeg [77]	[ICLR'21]	R101	61.0	–	69.8	72.4	–
CPS [7]	[CVPR'21]	R101	–	69.8	74.3	74.6	76.8
U²PL [60]	[CVPR'22]	R101	–	74.9	76.5	78.5	79.1
PCR [62]	[NeurIPS'22]	R101	–	73.4	76.3	78.4	79.1
AEL [31]	[NeurIPS'21]	R101	–	75.8	77.9	79.0	80.3
ESL [43]	[ICCV'23]	R101	–	75.1	77.2	78.9	80.5
3-CPS [37]	[ICCV'23]	R101	–	75.7	77.4	78.5	–
UniMatch [68]	[CVPR'23]	R101	73.0	76.6	77.9	79.2	79.5
LogicDiag [38]	[ICCV'23]	R101	–	76.8	78.9	80.2	**81.0**
UniMatch† [68]	[CVPR'23]	ViT-B/16	73.8	76.6	78.2	79.1	79.6
Ours	—	ViT-B/16	**76.2**	**77.9**	**79.4**	**80.3**	80.6
			(+2.4)	(+1.3)	(+1.2)	(+1.2)	(+1.0)

4.2. Comparison with the State of the Art

We compare SemiVL with previous semi-supervised methods on four popular semantic segmentation datasets in Tab. 1-5. We additionally reproduce the SOTA method UniMatch [68] with a ViT [13] backbone and the same training parameters as SemiVL for a fair comparison. On all four datasets, SemiVL achieves consistent and significant gains over previous works as detailed in the following.

Pascal VOC: The Pascal VOC dataset [14] contains 10582 training images of which 1464 images have segmentation annotations for 21 classes. Tab. 1 compares SemiVL with previous methods for different portions of labeled images ranging from 1/115 of the training set (92 labels) up to 1/7 (1464 labels). SemiVL significantly improves the performance from +3.3 mIoU (1464 labels) up to +6.1 (92 labels) over UniMatch(ViT). The larger improvement for fewer annotations shows the efficiency of SemiVL to learn from limited annotations. SemiVL can reduce the need for labels by a factor of 8 compared to UniMatch(ViT). As a further strong baseline using CLIP, we integrate the open-vocabulary method ZegCLIP [76] into UniMatch. Even though it improves the performance over UniMatch, SemiVL still significantly outperforms it. The class-wise analysis in Tab. 2 shows that SemiVL particularly improves classes, where UniMatch struggles (low IoU), such as chair, dining table, or sofa. These are classes that are often confused with the background class or semantically similar classes. This is also reflected in the example predictions in Fig. 5.

COCO: The COCO dataset [4] has 118k training images with segmentation annotations for 81 classes. The higher number of classes makes it more challenging for semi-supervised learning. Here, the power of vision-language guidance to distinguish semantic concepts particularly shines. SemiVL achieves major mIoU gains of up to +13.5 (for 232 labels) over UniMatch(ViT) as shown in Tab. 3.

ADE20K: The ADE dataset [71] has 20210 training images with segmentation annotations for 150 classes and is even more challenging than COCO. Tab. 4 shows that SemiVL achieves gains of up to +9.7 mIoU (for 158 labels).

Cityscapes: The Cityscapes dataset [11] has 2975 training images of street scenes with fine annotations for 19 classes. Tab. 5 shows that SemiVL outperforms UniMatch+ViT by up to +2.4 mIoU (100 labels). We hypothesize that the improvements are smaller than for the other benchmarks due to many challenging small classes in Cityscapes (e.g. distant poles or traffic signs), which are usually not part of the captions used in vision-language pre-training, limiting the

Table 6. **Ablation study** of SemiVL's components on Pascal VOC: vision-language pre-training (VL Pretr.), spatial fine-tuning (SFT), language-guided decoder (Lang.Dec.), dense CLIP guidance (CLIP Guid.), and Class Definition Guidance (Cls.Def.).

VL Pretr.	SFT	Lang.Dec.	CLIP Guid.	Cls.Def.	$mIoU_{92}$		$mIoU_{1464}$	
–	–	–	–	–	77.9	↘	84.0	↘
✓	–	–	–	–	80.6	+2.7	86.0	+2.0
✓	✓	–	–	–	81.7	+3.8	86.7	+2.7
✓	✓	✓	–	–	82.7	+4.8	87.3	+3.3
✓	✓	✓	✓	–	83.2	+5.3	**87.4**	**+3.4**
✓	✓	✓	✓	✓	**84.0**	**+6.1**	87.3	+3.3

Table 7. Comparison of **fine-tuning strategies** on Pascal VOC.

Fine-Tuning Method	#Updated Params. (Encoder)	$mIoU_{92}$	$mIoU_{1464}$
Frozen	–	75.9	81.7
Deep Prompt Tuning [76]	0.7 M	79.5	84.3
LoRA [30]	1.4 M	80.5	85.8
Full Fine-Tuning	86.8 M	80.6	86.0
Spatial Fine-Tuning	29.1 M	**81.7**	**86.7**

Table 8. **Decoder ablation study** on Pascal VOC.

Decoder Variant	$mIoU_{92}$	$mIoU_{1464}$
Joint Reasoning	64.9	86.3
VL Similarity	68.6	84.5
VL S. + Up	74.1	86.6
VL S. + Up + Spatial	80.3	86.9
VL S. + Up + Spatial + Semantic	**82.7**	**87.3**

Table 9. Study on **class definition guidance** on VOC.

Class Definition	$mIoU_{92}$
Dictionary Def. (Text)	82.7
GPT Concepts (Max)	82.9
Class Name	83.2
Guidelines (Text)	83.4
Guidel. (Avg)	83.4
Ours: Guidel. (Max)	**84.0**

efficiency of vision-language guidance.

4.3. Analysis

Ablation Study: Tab. 6 shows the ablation of SemiVL on Pascal VOC. The first row shows the UniMatch baseline with an ImageNet pre-trained ViT encoder and DeepLabv3+ decoder. UniMatch uses consistency training (see Sec. 3.1). All components of SemiVL contribute to the major performance gain over UniMatch. The largest improvement originates from the VL pre-training, improving the semantic understanding. But also the spatial fine-tuning and the language-guided decoder significantly improve the performance for both 92 and 1464 labels. The CLIP guidance and the class definitions improve the mIoU for 92 labels, while they do not affect the mIoU for 1464 labels. Probably, the $16\times$ more labels provide sufficient information about the class definitions of the dataset so that no additional CLIP guidance is necessary in this case. All components provide stronger improvements for fewer labels, showing their suitability for training with limited annotations.

Fine-Tuning Strategies: Tab. 7 compares different methods to adapt the VLM to semantic segmentation. While a frozen backbone does not overfit to the small labeled subset, it also cannot adapt to segmentation. Deep prompt tuning (DPT) [75, 76] learns additional input tokens to manipulate a frozen ViT. We use the same parameters as [76]. It significantly improves over a frozen backbone. LoRA [30] learns low-rank ($r=8$) residuals for the attention weights (W_{qkvo}). It improves over DPT achieving a similar performance as full model fine-tuning. Our spatial fine-tuning can further improve over both. On the one side, it preserves the semantic reasoning of the local MLPs and avoids their overfitting to the limited labels. On the other side, its full rank fine-tuning of the attention layers might be necessary to sufficiently adapt the ViT from image-level to dense reasoning. For all variants, we tuned the backbone learning rate.

Decoder Study: SemiVL's decoder is designed to decouple semantic from spatial reasoning for label-efficient learning. As a baseline for joint reasoning, we input all classes together into the spatial reasoning (N times more convolution channels) instead of processing each class separately (and sharing the parameters across classes). Tab. 8 shows that decoupled reasoning (82.7 mIoU) significantly outperforms joint reasoning (64.9 mIoU) for 92 labels as the joint reasoning overfits to the small labeled subset. Further, Tab. 8 provides an ablation study over SemiVL's decoder modules. It shows that each module (upsampling, spatial reasoning, and semantic reasoning) is crucial for its performance.

Class Definitions: Tab. 9 compares different strategies for text prompts in CLIP inference. It shows that class definitions from the dataset annotation guidelines achieve the best performance. Here, taking the maximum score over concepts (Sec. 3.6) works better than using an averaged text embedding over the concepts or the guidelines as raw text. As baselines, we also benchmark definitions from the Oxford Languages dictionary and class concepts queried from GPT3.5. However, they perform worse, probably because they are dataset agnostic and do not represent the dataset-specific class boundaries as well as the annotation guidelines.

5. Conclusions

In this work, we have investigated the impact of vision-language guidance on semi-supervised semantic segmentation. First, we have shown that VL pre-training is a particularly suited strategy with fine-grained semantic understanding. Second, we introduced a spatial fine-tuning strategy to semi-supervised learning to efficiently adapt VL pre-training from image-level to dense understanding. Third, we demonstrated the power of vision-language alignment in a specifically designed decoder architecture. And fourth, we regularized the training with guidance from a frozen CLIP model with class definition guidelines. Our SemiVL framework combines the strategies and provides major performance improvements over previous state-of-the-art methods on multiple benchmarks. Overall, SemiVL preserves performance using only a quarter of the annotated images.

References

[1] Eric Arazo, Diego Ortego, Paul Albert, Noel E O'Connor, and Kevin McGuinness. Pseudo-labeling and confirmation bias in deep semi-supervised learning. In *International Joint Conference on Neural Networks*, pages 1–8. IEEE, 2020. 2

[2] Philip Bachman, Ouais Alsharif, and Doina Precup. Learning with pseudo-ensembles. *NeurIPS*, 27, 2014. 3

[3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *NeurIPS*, 33:1877–1901, 2020. 3

[4] Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Cocostuff: Thing and stuff classes in context. In *CVPR*, pages 1209–1218, 2018. 6, 7

[5] Junbum Cha, Jonghwan Mun, and Byungseok Roh. Learning to generate text-grounded mask for open-world semantic segmentation from only image-text pairs. In *CVPR*, pages 11165–11174, 2023. 2, 3

[6] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *ECCV*, pages 801–818, 2018. 5, 6

[7] Xiaokang Chen, Yuhui Yuan, Gang Zeng, and Jingdong Wang. Semi-supervised semantic segmentation with cross pseudo supervision. In *CVPR*, pages 2613–2622, 2021. 3, 4, 6, 7

[8] Xi Chen, Xiao Wang, Lucas Beyer, Alexander Kolesnikov, Jialin Wu, Paul Voigtlaender, Basil Mustafa, Sebastian Goodman, Ibrahim Alabdulmohsin, Piotr Padlewski, et al. Pali-3 vision language models: Smaller, faster, stronger. *arXiv preprint arXiv:2310.09199*, 2023. 3

[9] Seokju Cho, Heeseong Shin, Sunghwan Hong, Seungjun An, Seungjun Lee, Anurag Arnab, Paul Hongsuck Seo, and Seungryong Kim. Cat-seg: Cost aggregation for open-vocabulary semantic segmentation. *arXiv preprint arXiv:2303.11797*, 2023. 3, 4

[10] MMSegmentation Contributors. MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark. https://github.com/open-mmlab/mmsegmentation, 2020. 12

[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 *CVPR*, pages 3213–3223, 2016. 6, 7

[12] Jian Ding, Nan Xue, Gui-Song Xia, and Dengxin Dai. Decoupling zero-shot semantic segmentation. In *CVPR*, pages 11583–11592, 2022. 2, 3

[13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 4, 6, 7

[14] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88:303–338, 2010. 6, 7

[15] Zhengyang Feng, Qianyu Zhou, Qiqi Gu, Xin Tan, Guangliang Cheng, Xuequan Lu, Jianping Shi, and Lizhuang Ma. Dmt: Dynamic mutual training for semi-supervised learning. *PR*, 130:108777, 2022. 2

[16] Geoffrey French, Samuli Laine, Timo Aila, Michal Mackiewicz, and Graham Finlayson. Semi-supervised semantic segmentation needs strong, varied perturbations. In *BMVC*, 2020. 3, 7

[17] Golnaz Ghiasi, Xiuye Gu, Yin Cui, and Tsung-Yi Lin. Scaling open-vocabulary image segmentation with image-level labels. In *ECCV*, pages 540–557. Springer, 2022. 3

[18] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *NeurIPS*, 27, 2014. 2

[19] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. *NeurIPS*, 17, 2004. 2

[20] Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. *ICLR*, 2021. 3

[21] Dayan Guan, Jiaxing Huang, Aoran Xiao, and Shijian Lu. Unbiased subclass regularization for semi-supervised semantic segmentation. In *CVPR*, pages 9968–9978, 2022. 3

[22] Ruifei He, Jihan Yang, and Xiaojuan Qi. Re-distributing biased pseudo labels for semi-supervised semantic segmentation: A baseline investigation. In *ICCV*, pages 6930–6940, 2021. 3

[23] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *ICML*, pages 2790–2799. PMLR, 2019. 3

[24] 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 *CVPR*, pages 11130–11140, 2021. 3

[25] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. DAFormer: Improving network architectures and training strategies for domain-adaptive semantic segmentation. In *CVPR*, pages 9924–9935, 2022. 3

[26] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. HRDA: Context-aware high-resolution domain-adaptive semantic segmentation. In *ECCV*, pages 372–391. Springer, 2022. 3

[27] Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Domain adaptive and generalizable network architectures and training strategies for semantic image segmentation. *IEEE TPAMI*, 2023. 3

[28] Lukas Hoyer, Dengxin Dai, Haoran Wang, and Luc Van Gool. MIC: Masked image consistency for context-enhanced domain adaptation. In *CVPR*, pages 11721–11732, 2023. 3

[29] Lukas Hoyer, Dengxin Dai, Qin Wang, Yuhua Chen, and Luc Van Gool. Improving semi-supervised and domain-adaptive semantic segmentation with self-supervised depth estimation. *IJCV*, pages 1–27, 2023. 3

[30] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *ICLR*, 2022. 3, 4, 8

[31] Hanzhe Hu, Fangyun Wei, Han Hu, Qiwei Ye, Jinshi Cui, and Liwei Wang. Semi-supervised semantic segmentation via adaptive equalization learning. *NeurIPS*, 34:22106–22118, 2021. 3, 7

[32] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *ECCV*, pages 709–727. Springer, 2022. 3, 4

[33] Muhammad Gul Zain Ali Khan, Muhammad Ferjad Naeem, Luc Van Gool, Didier Stricker, Federico Tombari, and Muhammad Zeshan Afzal. Introducing language guidance in prompt-based continual learning. In *ICCV*, pages 11463–11473, 2023. 3

[34] Xin Lai, Zhuotao Tian, Li Jiang, Shu Liu, Hengshuang Zhao, Liwei Wang, and Jiaya Jia. Semi-supervised semantic segmentation with directional context-aware consistency. In *CVPR*, pages 1205–1214, 2021. 3

[35] Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *ICML Workshop on challenges in representation learning*, page 896. Atlanta, 2013. 2

[36] Boyi Li, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and Rene Ranftl. Language-driven semantic segmentation. In *ICLR*, 2022. 3

[37] Yijiang Li, Xinjiang Wang, Lihe Yang, Litong Feng, Wayne Zhang, and Ying Gao. Diverse cotraining makes strong semi-supervised segmentor. In *ICCV*, pages 16055–16067, 2023. 7

[38] Chen Liang, Wenguan Wang, Jiaxu Miao, and Yi Yang. Logic-induced diagnostic reasoning for semi-supervised semantic segmentation. In *ICCV*, pages 16197–16208, 2023. 3, 7

[39] Feng Liang, Bichen Wu, Xiaoliang Dai, Kunpeng Li, Yinan Zhao, Hang Zhang, Peizhao Zhang, Peter Vajda, and Diana Marculescu. Open-vocabulary semantic segmentation with mask-adapted clip. In *CVPR*, pages 7061–7070, 2023. 3

[40] Yuqi Lin, Minghao Chen, Wenxiao Wang, Boxi Wu, Ke Li, Binbin Lin, Haifeng Liu, and Xiaofei He. Clip is also an efficient segmenter: A text-driven approach for weakly supervised semantic segmentation. In *CVPR*, pages 15305–15314, 2023. 6

[41] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 6

[42] Huaishao Luo, Junwei Bao, Youzheng Wu, Xiaodong He, and Tianrui Li. Segclip: Patch aggregation with learnable centers for open-vocabulary semantic segmentation. In *ICML*, pages 23033–23044. PMLR, 2023. 3

[43] Jie Ma, Chuan Wang, Yang Liu, Liang Lin, and Guanbin Li. Enhanced soft label for semi-supervised semantic segmentation. In *ICCV*, pages 1185–1195, 2023. 3, 7

[44] Sudhanshu Mittal, Maxim Tatarchenko, and Thomas Brox. Semi-supervised semantic segmentation with high-and low-level consistency. *IEEE TPAMI*, 43(4):1369–1379, 2019. 1, 2

[45] Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The role of context for object detection and

[46] Muhammad Ferjad Naeem, Yongqin Xian, Luc V Gool, and Federico Tombari. I2dformer: Learning image to document attention for zero-shot image classification. *NeurIPS*, 2022. 3

[47] Muhammad Ferjad Naeem, Muhammad Gul Zain Ali Khan, Yongqin Xian, Muhammad Zeshan Afzal, Didier Stricker, Luc Van Gool, and Federico Tombari. I2mvformer: Large language model generated multi-view document supervision for zero-shot image classification. In *CVPR*, 2023. 3, 6

[48] Muhammad Ferjad Naeem, Yongqin Xian, Xiaohua Zhai, Lukas Hoyer, Luc Van Gool, and Federico Tombari. Silc: Improving vision language pretraining with self-distillation. *arXiv preprint arXiv:2310.13355*, 2023. 3

[49] Viktor Olsson, Wilhelm Tranheden, Juliano Pinto, and Lennart Svensson. Classmix: Segmentation-based data augmentation for semi-supervised learning. In *WACV*, pages 1369–1378, 2021. 3

[50] Sarah Pratt, Ian Covert, Rosanne Liu, and Ali Farhadi. What does a platypus look like? generating customized prompts for zero-shot image classification. In *ICCV*, pages 15691–15701, 2023. 3, 6

[51] 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*, pages 8748–8763. PMLR, 2021. 1, 3, 4, 6

[52] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015. 5

[53] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. *NeurIPS*, 29, 2016. 3

[54] Gyungin Shin, Weidi Xie, and Samuel Albanie. Reco: Retrieve and co-segment for zero-shot transfer. In *NeurIPS*, 2022. 3

[55] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk, Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. *NeurIPS*, 33:596–608, 2020. 1, 3, 4

[56] Nasim Souly, Concetto Spampinato, and Mubarak Shah. Semi supervised semantic segmentation using generative adversarial network. In *ICCV*, pages 5688–5696, 2017. 1, 2

[57] Yi-Lin Sung, Jaemin Cho, and Mohit Bansal. Vl-adapter: Parameter-efficient transfer learning for vision-and-language tasks. In *CVPR*, pages 5227–5237, 2022. 3, 4

[58] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *NeurIPS*, 30, 2017. 5, 6

[59] Can Wang, Menglei Chai, Mingming He, Dongdong Chen, and Jing Liao. Clip-nerf: Text-and-image driven manipulation of neural radiance fields. In *CVPR*, pages 3835–3844, 2022. 3

semantic segmentation in the wild. In *CVPR*, pages 891–898, 2014. 13

[60] Yuchao Wang, Haochen Wang, Yujun Shen, Jingjing Fei, Wei Li, Guoqiang Jin, Liwei Wu, Rui Zhao, and Xinyi Le. Semi-supervised semantic segmentation using unreliable pseudo-labels. In *CVPR*, pages 4248–4257, 2022. 3, 6, 7

[61] Yuxin Wu and Kaiming He. Group normalization. In *ECCV*, pages 3–19, 2018. 6

[62] Haiming Xu, Lingqiao Liu, Qiuchen Bian, and Zhen Yang. Semi-supervised semantic segmentation with prototype-based consistency regularization. *NeurIPS*, 35:26007–26020, 2022. 7

[63] Jiarui Xu, Shalini De Mello, Sifei Liu, Wonmin Byeon, Thomas Breuel, Jan Kautz, and Xiaolong Wang. Groupvit: Semantic segmentation emerges from text supervision. In *CVPR*, 2022. 2, 3

[64] Mengde Xu, Zheng Zhang, Fangyun Wei, Yutong Lin, Yue Cao, Han Hu, and Xiang Bai. A simple baseline for open-vocabulary semantic segmentation with pre-trained vision-language model. In *ECCV*, pages 736–753. Springer, 2022. 3

[65] Mengde Xu, Zheng Zhang, Fangyun Wei, Han Hu, and Xiang Bai. Side adapter network for open-vocabulary semantic segmentation. In *CVPR*, pages 2945–2954, 2023. 2, 3

[66] Yi Xu, Lei Shang, Jinxing Ye, Qi Qian, Yu-Feng Li, Baigui Sun, Hao Li, and Rong Jin. Dash: Semi-supervised learning with dynamic thresholding. In *ICML*, pages 11525–11536. PMLR, 2021. 3

[67] Lihe Yang, Wei Zhuo, Lei Qi, Yinghuan Shi, and Yang Gao. St++: Make self-training work better for semi-supervised semantic segmentation. In *CVPR*, pages 4268–4277, 2022. 1, 2, 3, 7

[68] Lihe Yang, Lei Qi, Litong Feng, Wayne Zhang, and Yinghuan Shi. Revisiting weak-to-strong consistency in semi-supervised semantic segmentation. In *CVPR*, pages 7236–7246, 2023. 1, 2, 3, 4, 6, 7, 12, 14, 15, 16, 17, 18

[69] Bowen Zhang, Yidong Wang, Wenxin Hou, Hao Wu, Jindong Wang, Manabu Okumura, and Takahiro Shinozaki. Flex-match: Boosting semi-supervised learning with curriculum pseudo labeling. *NeurIPS*, 34:18408–18419, 2021. 3

[70] Yuanyi Zhong, Bodi Yuan, Hong Wu, Zhiqiang Yuan, Jian Peng, and Yu-Xiong Wang. Pixel contrastive-consistent semi-supervised semantic segmentation. In *ICCV*, pages 7273–7282, 2021. 7

[71] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *CVPR*, pages 633–641, 2017. 6, 7

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

[73] Chong Zhou, Chen Change Loy, and Bo Dai. Extract free dense labels from clip. In *ECCV*, pages 696–712. Springer, 2022. 1, 2, 3, 4, 6, 12

[74] Jinghao Zhou, Li Dong, Zhe Gan, Lijuan Wang, and Furu Wei. Non-contrastive learning meets language-image pre-training. In *CVPR*, 2023. 3

[75] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *IJCV*, 130(9):2337–2348, 2022. 3, 8

[76] Ziqin Zhou, Yinjie Lei, Bowen Zhang, Lingqiao Liu, and Yifan Liu. Zegclip: Towards adapting clip for zero-shot semantic segmentation. In *CVPR*, pages 11175–11185, 2023. 2, 3, 7, 8, 12

[77] Yuliang Zou, Zizhao Zhang, Han Zhang, Chun-Liang Li, Xiao Bian, Jia-Bin Huang, and Tomas Pfister. Pseudoseg: Designing pseudo labels for semantic segmentation. In *ICLR*, 2020. 6, 7

SemiVL: Semi-Supervised Semantic Segmentation with Vision-Language Guidance

Supplementary Material

A. Overview

The supplementary material of SemiVL provides the source code (Sec. B), studies the influence of the hyperparameters of the dense CLIP guidance (Sec. C), provides details on the used class definitions (Sec. D), and qualitatively analyzes SemiVL on four semantic segmentation datasets (Pascal VOC, COCO, ADE20K, and Cityscapes) including a component ablation on Pascal VOC (Sec. E).

B. Source Code

The source code of SemiVL is provided at https://github.com/google-research/semivl. For further information on the environment setup and experiment execution, please refer to README.md. SemiVL is implemented in PyTorch. It is based on the implementations of UniMatch [68], MaskCLIP [73], ZegCLIP [76], and MMSegmentation [10].

C. Dense CLIP Guidance Hyperparameters

Tab. S1 shows the influence of the hyperparameters of the dense CLIP guidance loss on unlabeled images, i.e. the initial loss weight $\lambda_{DC,0}$ and the confidence threshold ζ. The optimal values are $\lambda_{DC,0} = 0.1$ and $\zeta = 0.9$. When $\lambda_{DC,0}$ is chosen smaller, the performance gradually decreases. However, a larger $\lambda_{DC,0}$ causes a considerable drop. Probably, the induced gradients from erroneous dense CLIP pseudo-labels become too strong and corrupt the default consistency training. Considering the confidence threshold ζ, the performance gradually degrades around the optimal value. Only when it is chosen too small and less confident CLIP predictions are included, the performance drops below the baseline.

Table S1. Study on the **hyperparameters of CLIP guidance** on Pascal VOC.

$\lambda_{DC,0}$	ζ	mIoU$_{92}$
0	0.9	82.7
0.01	0.9	83.6
0.1	0.9	**84.0**
1	0.9	75.7
0.1	0.7	81.5
0.1	0.8	83.2
0.1	0.9	**84.0**
0.1	0.99	83.9

Table S2. Class definitions from **annotation guidelines as class concepts (ours)** on Pascal VOC.

Class c	Concepts b_c
background	"background", "bed", "building", "cabinet", "ceiling", "curtain", "door", "fence", "floor", "grass", "ground", "mountain", "road", "rock", "shelves", "sidewalk", "sky", "snow", "tree", "wall", "water", "window", "hang glider", "helicopter", "jet ski", "go-cart", "tractor", "emergency vehicle", "lorry", "truck", "lion", "stool", "bench", "wheelchair", "coffee table", "desk", "side table", "picnic bench", "wolve", "flowers in a vase", "goat", "tram", "laptop", "advertising display", "vehicle interior"
aeroplane	"aeroplane", "airplane", "glider"
bicycle	"bicycle", "tricycle", "unicycle"
bird	"bird"
boat	"boat", "ship", "rowing boat", "pedalo"
bottle	"bottle", "plastic bottle", "glass bottle", "feeding bottle"
bus	"bus", "minibus"
car	"car", "van", "large family car", "realistic toy car"
cat	"cat", "domestic cat"
chair	"chair", "armchair", "deckchair"
cow	"cow"
dining table	"dining table", "table for eating at"
dog	"dog", "domestic dog"
horse	"horse", "pony", "donkey", "mule"
motorbike	"motorbike", "moped", "scooter", "sidecar"
person	"person", "people", "baby", "face"
potted plant	"potted plant", "indoor plant in a pot", "outdoor plant in a pot"
sheep	"sheep"
sofa	"sofa"
train	"train", "train carriage"
tv/monitor	"tv", "monitor", "standalone screen"

Table S3. Class definitions from **annotation guidelines as class concepts (ours)** on Cityscapes.

Class c	Concepts b_c
road	"road", "street", "parking space"
sidewalk	"sidewalk"
building	"building", "skyscaper", "house", "bus stop building", "garage", "car port", "scaffolding"
wall	"individual standing wall, which is not part of a building"
fence	"fence", "hole in fence"
pole	"pole", "sign pole", "traffic light pole"
traffic light	"traffic light"
traffic sign	"traffic sign", "parking sign", "direction sign"
vegetation	"vegetation", "tree", "hedge"
terrain	"terrain", "grass", "soil", "sand"
sky	"sky"
person	"person", "pedestrian", "walking person", "standing person", "person sitting on the ground", "person sitting on a bench", "person sitting on a chair"
rider	"rider", "cyclist", "motorcyclist"
car	"car", "jeep", "SUV", "van"
truck	"truck", "box truck", "pickup truck", "truck trailer"
bus	"bus"
train	"train", "tram"
motorcycle	"motorcycle", "moped", "scooter"
bicycle	"bicycle"

Table S4. Class definitions from **annotation guidelines (raw text)** on Pascal VOC.

Class c	Annotation Guidelines
background	"background"
aeroplane	"aeorplane including gliders but not hang gliders or helicopters"
bicycle	"bicycle including tricycles, unicycles"
bird	"bird"
boat	"boat including ships, rowing boats, pedaloes but not jet skis"
bottle	"bottle including plastic, glass or feeding bottles"
bus	"bus including minibus but not trams"
car	"car including vans, large family cars for 6-8 people, toy cars but not go-carts, tractors, emergency vehicles, lorries, trucks, or the vehicle interior"
cat	"domestic cat"
chair	"chair including armchairs, deckchairs, but not stools, wheelchairs, seats in buses or cars"
cow	"cow"
dining table	"table for eating at but not coffee tables, desks, side tables or picnic benches"
dog	"domestic dog (not wolves etc.)"
horse	"horse including ponies, donkeys, mules etc."
motorbike	"motorbike including mopeds, scooters, sidecars"
person	"person including babies, faces (i.e. truncated people)"
potted plant	"'indoor plants or outdoor plants clearly in a pot but not flowers in vases"
sheep	"sheep but not a goat"
sofa	"sofa excluding sofas made up as sofa-beds"
train	"train including train carriages, excluding trams"
tv/monitor	"tv/monitor including standalone screens but not laptops nor advertising displays"

Table S5. Class definitions as **concepts from GPT** on Pascal VOC.

Class c	GPT Concepts
background	"background", "scene", "environment", "setting", "context"
aeroplane	"aeroplane", "aircraft", "plane", "jet", "aviation"
bicycle	"bicycle", "bike", "cycle", "pedal", "two-wheeler"
bird	"bird", "avian", "feathered creature", "fowl", "winged animal"
boat	"boat", "vessel", "watercraft", "ship", "sailboat"
bottle	"bottle", "flask", "container", "jar", "vial"
bus	"bus", "coach", "transit", "shuttle", "public transport"
car	"car", "automobile", "vehicle", "motorcar", "sedan"
cat	"cat", "feline", "kitty", "kitten", "pussycat"
chair	"chair", "seat", "furniture", "stool", "armchair"
cow	"cow", "bovine", "cattle", "ox", "livestock"
diningtable	"diningtable", "table", "dining furniture", "dinner table", "kitchen table"
dog	"dog", "canine", "pooch", "puppy", "man's best friend"
horse	"horse", "equine", "stallion", "pony", "mare"
motorbike	"motorbike", "motorcycle", "bike", "motor", "two-wheeled vehicle"
person	"person", "human", "individual", "human being", "someone"
pottedplant	"pottedplant", "pot plant", "houseplant", "potted flower", "indoor plant"
sheep	"sheep", "lamb", "ewe", "ram", "woolly animal"
sofa	"sofa", "couch", "settee", "divan", "lounge"
train	"train", "locomotive", "railway vehicle", "railroad train", "engine"
tv/monitor	"tv/monitor", "television", "screen", "display", "monitor"

D. Class Definitions

In the following, we provide the used class definitions and their split into concepts. The class definitions are based on the annotation guidelines of Pascal VOC[2] and Cityscapes[3]. For COCO and ADE20K, we only use the class names as there are no annotation guidelines publicly available. For

Table S6. Class definition as **definitions from Oxford Languages** on Pascal VOC.

Class c	Oxford Languages Definition
background	"background"
aeroplane	"aeroplane", "a flying vehicle with fixed wings"
bicycle	"bicycle", "a vehicle consisting of two wheels held in a frame one behind the other, propelled by pedals and steered with handlebars attached to the front wheel"
bird	"bird", "a warm-blooded egg-laying vertebrate animal distinguished by the possession of feathers, wings, a beak, and typically by being able to fly"
boat	"boat", "a vessel for travelling over water, propelled by oars, sails, or an engine"
bottle	"bottle", "a glass or plastic container with a narrow neck, used for storing drinks or other liquids"
bus	"bus", "a large motor vehicle carrying passengers by road"
car	"car", "a four-wheeled road vehicle that is powered by an engine and is able to carry a small number of people"
cat	"cat", "a small domesticated carnivorous mammal with soft fur, a short snout, and retractable claws"
chair	"chair", "a separate seat for one person, typically with a back and four legs"
cow	"cow", "a fully grown female animal of a domesticated breed of ox, kept to produce milk or beef"
dining table	"dining table", "a table on which meals are served in a dining room"
dog	"dog", "a domesticated carnivorous mammal that typically has a long snout and non-retractable claws"
horse	"horse", "a large plant-eating domesticated mammal with solid hoofs and a flowing mane and tail, used for riding, racing, and to carry and pull loads"
motorbike	"motorbike", "a two-wheeled vehicle that is powered by a motor and has no pedals"
person	"person", "a human being regarded as an individual"
potted plant	"potted plant", "a plant in a pot"
sheep	"sheep", "a domesticated ruminant mammal with a thick woolly coat"
sofa	"sofa", "a long upholstered seat with a back and arms, for two or more people"
train	"train", "a series of connected railway carriages or wagons moved by a locomotive or by integral motors"
tv/monitor	"tv/monitor", "a device for watching television"

SemiVL, these free text class definitions are split into concepts b, which are assigned to the corresponding class c as described in the main paper. The resulting class concepts are shown in Tab. S2 for Pascal VOC and in Tab. S3 for Cityscapes. For, Pascal VOC we further add the background class names from the Pascal Context [45] set to $b_{background}$.

For the class definition study in the main paper, we further use the annotation guidelines as raw text (see Tab. S4), the definitions from the Oxford Languages dictionary (Tab. S6), and class concepts obtained from GTP3.5[4] (Tab. S5). In particular, it can be seen that by not knowing the dataset-specific decision boundaries, GPT concepts are not well aligned with the used dataset. For example, GPT produces "stool" as a concept for the class *chair*, while "stool" is considered *background* in Pascal VOC. Similar problems apply for "container", "jar", "furniture", and "table" (too generic).

[4] Prompt: "I have the following classes from a semantic segmentation dataset: ['background', 'aeroplane', 'bicycle', 'bird', 'boat', 'bottle', 'bus', 'car', 'cat', 'chair', 'cow', 'dining table', 'dog', 'horse', 'motorbike', 'person', 'potted plant', 'sheep', 'sofa', 'train', 'tv/monitor']. Please, provide 5 distinctive subcategories or synonyms for each class so that a vision language model can distinguish it from the other classes. None of these words should cause confusion with other classes. These words will be used in the prompt "a photo of a X". Please, provide the output as a nested python list."

E. Extended Qualitative Comparison

We present an extended qualitative comparison of SemiVL with UniMatch [68] in Fig. S1 for Pascal VOC with 92 labels, in Fig. S2 for COCO with 232 labels, in Fig. S3 for ADE20K with 158 labels, and in Fig. S4 for Cityscapes with 186 labels. They consistently show that SemiVL better distinguishes classes with a similar visual appearance. For Pascal VOC, Fig. S1 shows that SemiVL better distinguishes different animals (sheep, cat, dog, cow, and horse), furniture (chair, dining table, and sofa), and vehicles (bus, car, boat, airplane, and train). It also improves the distinction of foreground classes vs. the background class. For COCO, Fig. S2 shows that SemiVL better recognizes different animals (bear, sheep, cat, and cow), food (cake, donut, sandwich, and apple), furniture (couch, chair, book, and vase), and sports gear (skateboard, skis, kite, umbrella, and surfboard). For ADE20K, Fig. S3 shows that SemiVL better differentiates structures (tower, bridge, building, house, skyscraper, column, and wall), furniture (cabinet, door, chair, seat, table, sofa, and pool table), and ground types (rock, mountain, water, fountain, floor, rug, grass, and river). And for Cityscapes, Fig. S4 shows that SemiVL better distinguishes different vehicles (car, truck, bus, and train), different structures (fence, wall, and building), and ground types (road and sidewalk). To visualize them in a grid, all examples are the center crop of the full-sized predictions.

For deeper insights on the behavior of SemiVL's components, we visualize qualitative examples from the ablation study on VOC with 92 labels. Fig. S5 shows that each of SemiVL's components enhances its ability to distinguish semantically similar classes such as bicycle/motorbike, sofa/chair, car/motorbike, dining table/chair, bottle/vase(background), sofa/bed(background), and car/tractor(background). Specifically, VL pre-training and spatial fine-tuning exploit the rich semantic priors of CLIP, better recognizing the bicycle, sofa, and car. The language-guided decoder improves the spatial reasoning between object parts, enhancing the recognition of the chair/sofa seats from their context, i.e., the correctly classified chair legs and sofa armrests. The dense CLIP guidance counters drift issues appearing with spatial fine-tuning such as for the sofa and bottle. Finally, the class definitions help in the last two challenging rows as bed and tractor were defined as background in the class definitions (see Tab. S2).
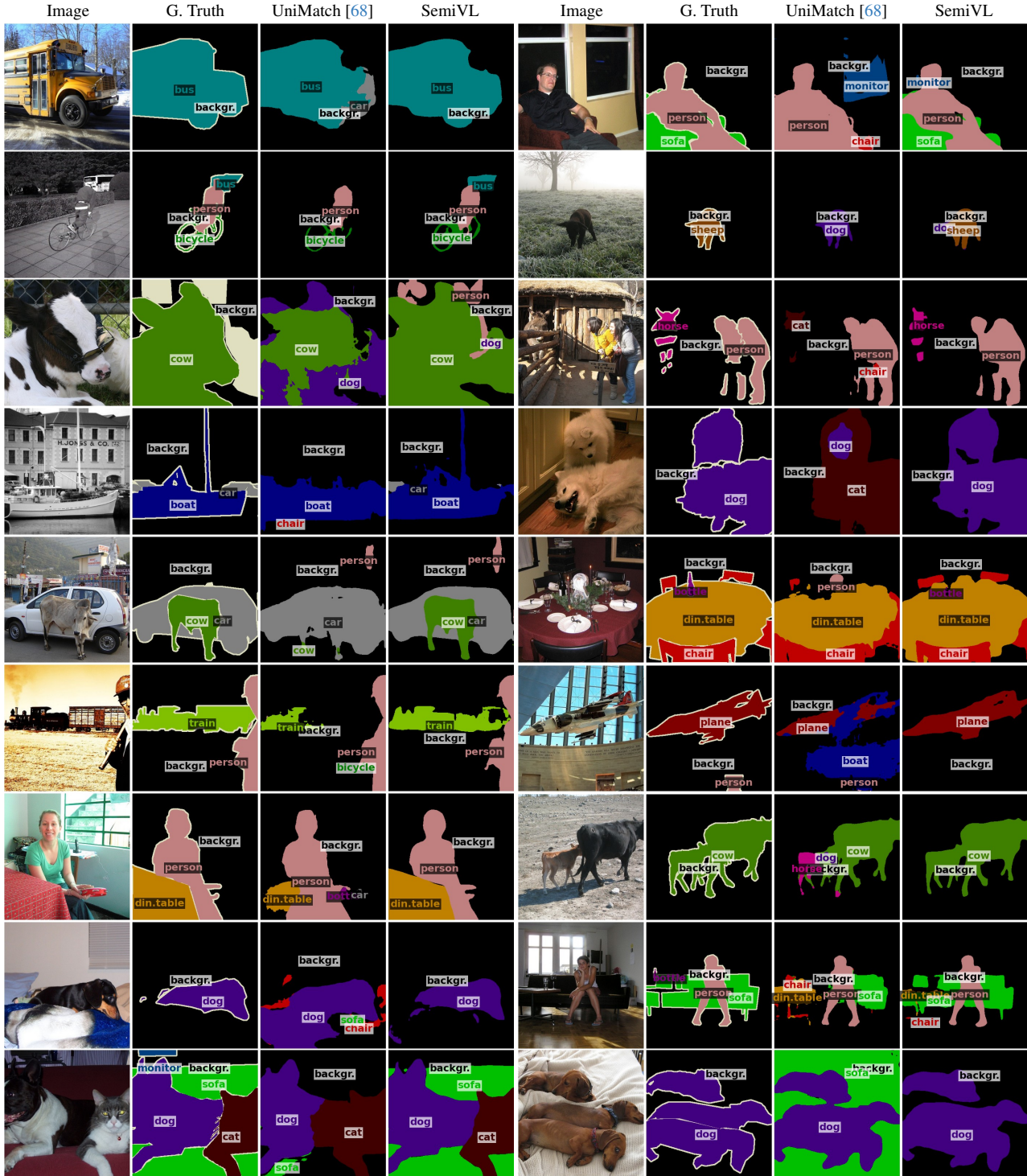
Figure S1. **Example predictions on VOC** (92 labels) showing the improved semantic understanding of SemiVL. In particular, SemiVL better distinguishes classes with similar visual appearance such as different animals (sheep, cat, dog, cow, and horse), different furniture (chair, dining table, and sofa), and different vehicles (bus, car, boat, airplane, and train). It also improves the distinction of foreground classes versus the background class.
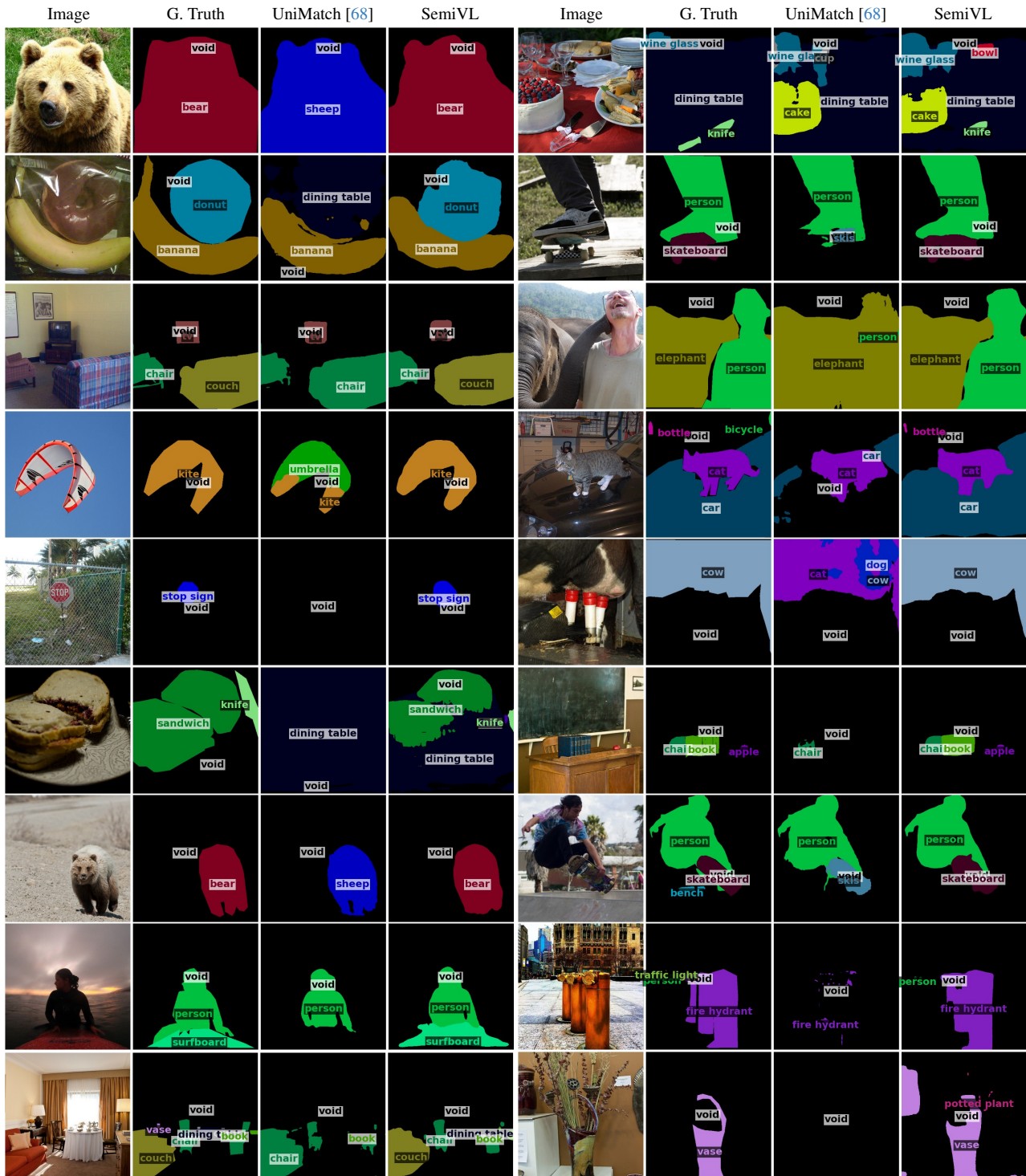
Figure S2. **Example predictions on COCO** (232 labels) showing the improved semantic understanding of SemiVL. In particular, SemiVL better distinguishes classes with similar visual appearance such as different animals (bear, sheep, cat, and cow), food (cake, donut, sandwich, and apple), furniture (couch, chair, book, and vase), and sports gear (skateboard, skis, kite, umbrella, and surfboard).

Figure S3. **Example predictions on ADE20K** (158 labels) showing the improved semantic understanding of SemiVL. In particular, SemiVL better distinguishes classes with similar visual appearance such as different structures (tower, bridge, building, house, skyscraper, column, and wall), different furniture (cabinet, door, chair, seat, table, sofa, and pool table), and ground types (rock, mountain, water, fountain, floor, rug, grass, and river).

Figure S4. **Example predictions on Cityscapes** (186 labels) showing the improved semantic understanding of SemiVL. In particular, SemiVL better distinguishes classes with similar visual appearance such as different vehicles (car, truck, bus, and train), different structures (fence, wall, and building), and ground types (road and sidewalk).
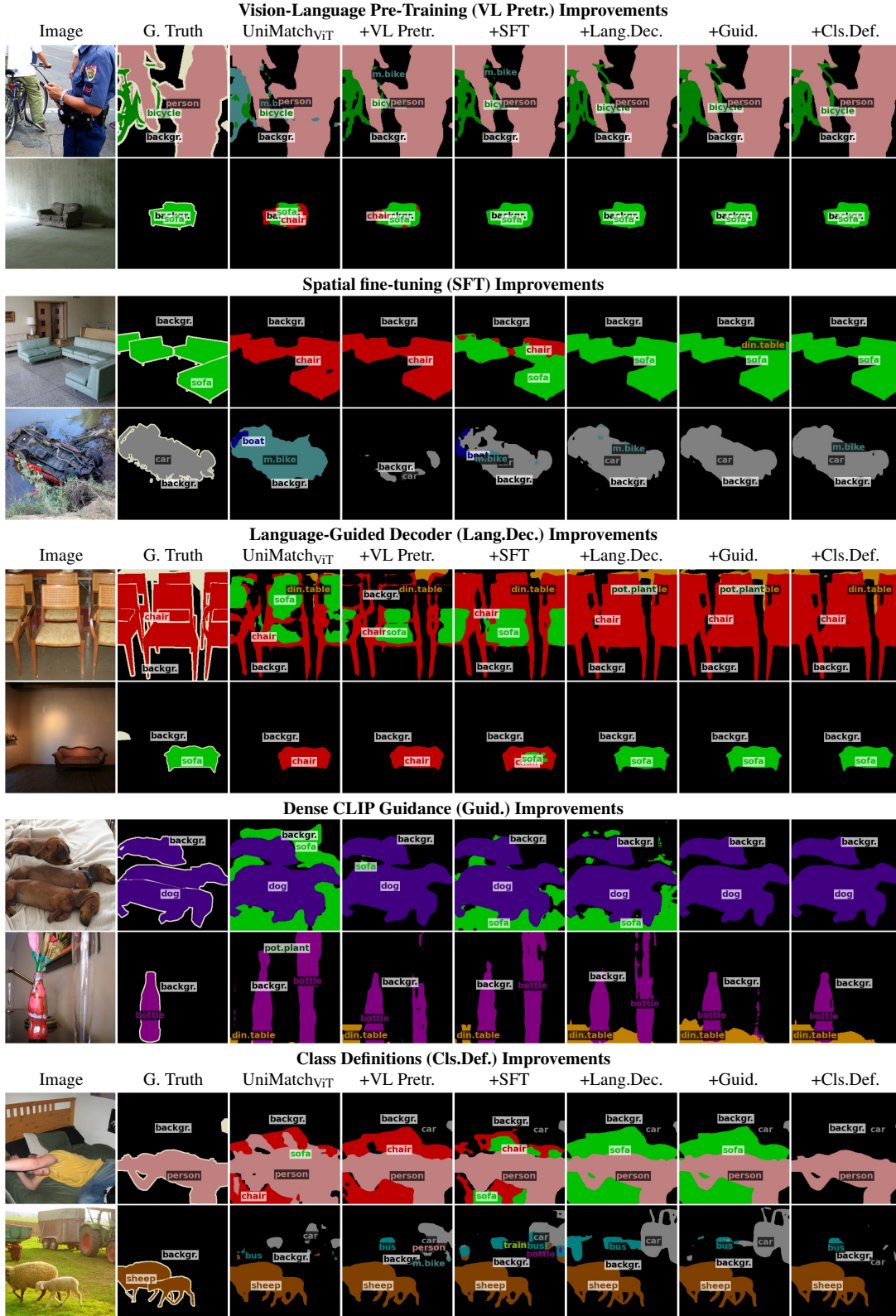
Figure S5. **Example predictions of the ablation study** on VOC (92 labels). Each of SemiVL's components enhances its ability to distinguish semantically similar classes such as bicycle/motorbike, sofa/chair, car/motorbike, dining table/chair, bottle/vase(background), sofa/bed(background), and car/tractor(background).