Supplementary Materials for Generative Semantic Segmentation

Jiaqi Chen[1] Jiachen Lu[1] Xiatian Zhu[2] Li Zhang[1*]
[1]Fudan University [2]University of Surrey

https://github.com/fudan-zvg/GSS

A. Proofs

A.1. Derivation of GSS ELBO

We provide the proof of Eq. (2) in the main paper here. We rewrite the log-likelihood of semantic segmentation $\log p(c|x)$ by introducing a discrete L-dimension latent distribution $q(z|c)$ (with $z \in \mathbb{Z}^L$).

$$\begin{aligned}\log p(c|x) &= \log \int p(c,z|x)\,\mathrm{d}z \\ &= \log \int p(c,z|x)\frac{q(z|c)}{q(z|c)}\,\mathrm{d}z \\ &= \log \mathbb{E}_{q(z|c)}\left[\frac{p(c,z|x)}{q(z|c)}\right] \\ &\geq \mathbb{E}_{q(z|c)}\left[\log \frac{p(c,z|x)}{q(z|c)}\right]\end{aligned}$$

(as $-log(\cdot)$ is convex, by Jensen's Inequality:

$f(\sum_i \lambda_i x_i) \leq \sum_i \lambda_i f(x_i)$, where $\lambda_i \geq 0, \sum_i \lambda_i = 1$)

$$\begin{aligned}&= \mathbb{E}_{q(z|c)}\left[\log \frac{p(c|z)p(z|x)}{q(z|c)}\right] \\ &= \mathbb{E}_{q(z|c)}\left[\log p(c|z)\right] + \mathbb{E}_{q(z|c)}\left[\log \frac{p(z|x)}{q(z|c)}\right] \\ &= \mathbb{E}_{q(z|c)}\left[\log p(c|z)\right] - D_{KL}(q(z|c), p(z|x)) \\ &= \mathbb{E}_{q_\phi(z|c)}\left[\log p_\theta(c|z)\right] - D_{KL}(q_\phi(z|c), p_\psi(z|x)).\end{aligned}$$

Different from ELBO [9] in VAE, the latent variable we introduce here is $q(z|c)$, rather than $q(z)$ to solve the conditioned mask generation problem.

A.2. Derivation of latent posterior learning

We provide the proof of Eq. (4) in the main paper here. As stated in main paper, the first stage latent posterior training is conducted by a MSE loss

$$\min_{\theta,\phi} \sum_c \mathbb{E}_{q_\phi(z|c)} \|p_\theta(c|z) - c\|.$$

*Li Zhang (lizhangfd@fudan.edu.cn) is the corresponding author with School of Data Science, Fudan University.

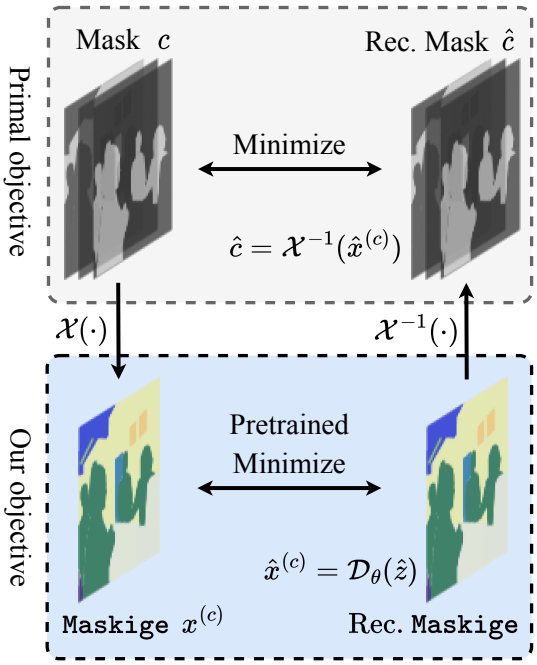

Figure S.1. **An illustration of our transformed objective.** `Rec.` stands for reconstruction.

Let us denote $\hat{c} = p_\theta(c|z)$ is the reconstructed mask. Then, we define a linear transform $x^{(c)} = \mathcal{X}_\beta(c) = c\beta$, where $\beta \in \mathbb{R}^{K\times 3}$ and an arbitrary inverse transform $\hat{c} = \mathcal{X}_\gamma^{-1}(\hat{x}^{(c)})$. Noted that the parameter γ can be non-linear. $x^{(c)}$ is called `maskige` and $\hat{x}^{(c)}$ is the reconstructed `maskige` produced by the `maskige` decoder $\hat{x}^{(c)} = \mathcal{D}_\theta(\hat{z})$. The transformed latent parameter $\hat{z}$ preserves the probability for the linear transformation,

$$q_{\hat{\phi}}(\hat{z}|x^{(c)}) = q_{\hat{\phi}}(\hat{z}|c\beta) = q_\phi(z|c). \tag{S.1}$$

Then, we have

$$\begin{aligned}&\min_{\theta,\phi} \sum_c \mathbb{E}_{q_\phi(z|c)} 2\|\hat{c} - c\| \\ &= \min_{\theta,\phi,\gamma} \sum_c \mathbb{E}_{q_\phi(z|c)} \left[\|\mathcal{X}^{-1}(\hat{x}^{(c)}) - c\| + \|\hat{c} - c\|\right].\end{aligned}$$

For the first term, since $\mathcal{X}^{-1}(\hat{x}^{(c)}) = \mathcal{X}^{-1}(\mathcal{D}_\theta(\hat{z}))$ which is not related to θ and ϕ. Therefore, we have

$$\begin{aligned}
&\min_{\theta,\phi,\gamma} \sum_c \mathbb{E}_{q_\phi(z|c)} \|\mathcal{X}^{-1}(\hat{x}^{(c)}) - c\| \\
=&\min_{\gamma} \sum_c \mathbb{E}_{q_\phi(z|c)} \|\mathcal{X}^{-1}(\mathcal{D}_\theta(\hat{z})) - c\| \\
=&\min_{\gamma,\beta} \sum_c \mathbb{E}_{q_{\hat{\phi}}(\hat{z}|x^{(c)})} \|\mathcal{X}^{-1}(\mathcal{D}_\theta(\hat{z})) - c\| \text{ (Eq. (S.1))}.
\end{aligned} \tag{S.2}$$

For the second term, $\hat{c} = \mathcal{D}_\theta(z)$ is not related to γ, which can be rewritten as

$$\begin{aligned}
&\min_{\theta,\phi} \sum_c \mathbb{E}_{q_\phi(z|c)} \|\hat{c} - c\| \\
=& \min_{\theta,\phi,s.t.\|\beta\|=1} \sum_c \mathbb{E}_{q_\phi(z|c)} \|\hat{c} - c\| \|\beta\| \\
=& \min_{\theta,\phi,s.t.\|\beta\|=1} \sum_c \mathbb{E}_{q_\phi(z|c)} \|\hat{c}\beta - c\beta\| \\
=& \min_{\theta,\phi,s.t.\|\beta\|=1} \sum_c \mathbb{E}_{q_\phi(z|c)} \|(\hat{c}\beta - \hat{x}^{(c)}) \text{ (equal to 0 by def.)} \\
&\qquad + (\hat{x}^{(c)} - x^{(c)}) \\
&\qquad + (x^{(c)} - c\beta)\| \text{ (equal to 0 by def.)} \\
=& \min_{\theta,\phi,s.t.\|\beta\|=1} \sum_{x^{(c)}} \mathbb{E}_{q_\phi(z|c)} \|\hat{x}^{(c)} - x^{(c)}\| \text{ (not related to } \theta) \\
=& \min_{\theta,\phi,s.t.\|\beta\|=1} \sum_{x^{(c)}} \mathbb{E}_{q_\phi(z|c)} \|\mathcal{D}_\theta(\hat{z}) - x^{(c)}\| \\
=& \min_{\theta,\hat{\phi},\beta,s.t.\|\beta\|=1} \sum_{x^{(c)}} \mathbb{E}_{q_{\hat{\phi}}(\hat{z}|x^{(c)})} \|\mathcal{D}_\theta(\hat{z}) - x^{(c)}\| \text{ (Eq. (S.1))}.
\end{aligned}$$

Combining Eq. (S.2) and Eq. (S.3), our final objective is

$$\begin{aligned}
&\min_{\hat{\phi},\theta,\beta,s.t.\|\beta\|=1} \sum_{x^{(c)}} \mathbb{E}_{q_{\hat{\phi}}(\hat{z}|x^{(c)})} \|\mathcal{D}_\theta(\hat{z}) - x^{(c)}\| \\
+ &\min_{\gamma,\beta} \sum_c \mathbb{E}_{q_{\hat{\phi}}(\hat{z}|x^{(c)})} \|\mathcal{X}^{-1}(\mathcal{D}_\theta(\hat{z})) - c\|.
\end{aligned} \tag{S.3}$$

For the first term, it is a VQVAE [18] reconstruction objective for `maskige`. Therefore, a VQVAE pretrained by DALL·E [16] with a large-scale OpenImage dataset can readily offer a good lower bound for the first term.

As such, only the second term is left for optimization. We can optimize the γ with gradient descent, corresponding to GSS-FT&TT. Besides, we can solve this problem more efficiently by a linear assumption, *i.e.* $\mathcal{X}^{-1}(\hat{x}^{(c)}) = \hat{x}^{(c)}\gamma$ where $\gamma \in \mathbb{R}^{3\times K}$. We denote the $\hat{X}^{(c)}$ is a matrix with each row an reconstructed `maskige` and C is a matrix with each row an input mask. We solve the optimization with least square error

$$\begin{aligned}
&\|\mathcal{X}^{-1}(\hat{X}^{(c)}) - C\|^2 \\
=&\|\hat{X}^{(c)}\gamma - C\|^2 \\
=&\|(\hat{X}^{(c)} - X^{(c)} + X^{(c)})\gamma - C\|^2 \\
\le& \left(\|\hat{X}^{(c)} - C\beta\|\|\gamma\| + \|X^{(c)}\gamma - C\|\right)^2.
\end{aligned} \tag{S.4}$$

The optimization over both β and γ is non-convex (as shown by the poor performance with GSS-TT), so we optimize them sequentially in GSS-FF&FT&TF. For GSS-FF, we use a hand-crafted optimized β.

$$\begin{aligned}
&\left(\|\hat{X}^{(c)} - C\beta\|\|\gamma\| + \|X^{(c)}\gamma - C\|\right)^2 \\
\le& \left(\tau\|\gamma\| + \|X^{(c)}\gamma - C\|\right)^2 \\
=& \left(\tau\|\gamma\| + \|C\beta\gamma - C\|\right)^2 \\
\le& \left(\tau\|\gamma\| + \|C\|\|\beta\gamma - \not{1}\|\right)^2.
\end{aligned} \tag{S.5}$$

where $\tau = \|\hat{X}^{(c)} - C\beta\|$ is bounded and unrelated to γ by provided VQVAE and β. Our objective then changes to minimize the upper bound.

$$\begin{aligned}
\min_{\gamma,s.t.\|\gamma\|=1} \text{RSS}(\gamma) &= \min_{\gamma,s.t.\|\gamma\|=1} \|\beta\gamma - \not{1}\|^2 \\
&= \min_{\gamma,s.t.\|\gamma\|=1} (\beta\gamma - \not{1})^\top(\beta\gamma - \not{1}).
\end{aligned} \tag{S.6}$$

We take the derivative of Eq. (S.6), then

$$\frac{\partial \text{RSS}}{\partial \gamma} = 2\beta^\top(\beta\gamma - \not{1}) = 0. \tag{S.7}$$

The unique solution of Eq. (S.7) is

$$\begin{aligned}
&\beta^\top\beta\gamma = \beta^\top \\
(\Rightarrow)\quad &(\beta^\top\beta)^{-1}(\beta^\top\beta)\gamma = (\beta^\top\beta)^{-1}\beta^\top \\
(\Rightarrow)\quad &\gamma = (\beta^\top\beta)^{-1}\beta^\top.
\end{aligned} \tag{S.8}$$

For the special design GSS-TF, we use a cascaded optimization to automatically optimize β and γ.

$$\begin{aligned}
\beta_{t+1} &= \arg\min_{\beta} \left(\|\hat{X}^{(c)} - C\beta\|\|\gamma_t\| + \|C\beta\gamma_t - C\|\right)^2 \\
\gamma_{t+1} &= (\beta_{t+1}^\top\beta_{t+1})^{-1}\beta_{t+1}^\top.
\end{aligned}$$

The β_{t+1} is optimized by one mini-batch step of gradient descent.

B. `Maskige` optimization designs

As illustrated above, β is a linear projection applied on the ground-truth mask, *i.e.* $x^{(c)} = c\beta$. As is shown in

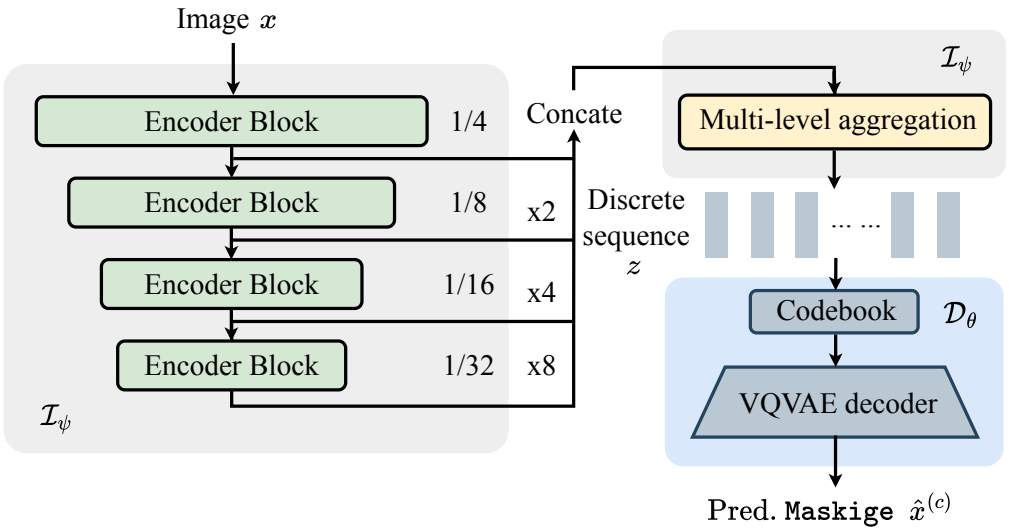

Figure S.2. **Encoder-decoder style architecture of GSS.** "Pred." stands for prediction.

Method	Iteration	VOC [7]	Context [15]	CamVid [1]	WildDash [21]	KITTI [8]	ScanNet [4]	*h. mean*
- Discriminative modeling:								
CCSA [14]	500k	48.9	-	52.4	36.0	-	27.0	39.7
MGDA [17]	500k	69.4	-	57.5	39.9	-	33.5	46.1
MSeg-w/o relabel [11]	500k	70.2	42.7	82.0	62.7	65.5	43.2	57.6
MSeg [11]	500k	70.7	42.7	83.3	62.0	67.0	48.2	59.2
MSeg-480p [11]	1,500k	76.4	45.9	81.2	62.7	**68.2**	49.5	61.2
MSeg-720p [11]	1,500k	74.7	44.0	83.5	60.4	67.9	47.7	59.8
MSeg-1080p [11]	1,500k	72.0	44.0	**84.5**	59.9	66.5	49.5	59.8
- Generative modeling:								
GSS-FF (Ours)	160k	78.7	45.8	74.2	61.8	65.4	46.9	59.5
GSS-FT-W (Ours)	160k	**79.5**	**47.7**	75.9	**65.3**	68.0	**49.7**	**61.9**

Table S.1. **Additional cross-domain semantic segmentation performance on MSeg dataset `test` split [11]:** We add performance of Mseg-480p, Mseg-720p and Mseg-1080p [11] results to Table 6 of the main paper. **No improved versions of our methods are included.** "480p", "720p" and "1080p" mean all test images are resized to 480p (the shorter side is 480 pixel), 720p and 1080p, respectively, when MSeg model inference.

Eq. (S.4) and Eq. (S.5), the quality of $\|\hat{X}^{(c)} - C\beta\|$ directly affects the quality of the following γ optimization, so the optimization of β will affect the difficulty degree of our learning of $\mathcal{X}_\gamma^{-1}$. The cascaded optimization in GSS-TF (84.37% reconstruction mIoU) provides an upper-bound for β optimization. However, the gradient descent optimization costs another 5 GPU hours according to Table 1 of main paper. Therefore, we propose a hand-crafted optimization of β in GSS-FF&FT&FT-W that achieves satisfying performance without requiring extra training time, based on the *maximal distance assumption*.

To understand this assumption, we can consider the linear projection parameter $\beta \in \mathbb{R}^{K\times 3}$ as a colorization process, where each category is assigned an `rgb` color. The idea behind the *maximal distance assumption* is to maximize the color difference between the encoding of each category. For instance, if two different categories are assigned similar colors, the model may struggle to differentiate between them. Therefore, by maximizing the distance between color embeddings, we can improve the model's ability to distinguish between categories. We interpret the parameter β as R, G, B color sequences $\mathcal{A}^r, \mathcal{A}^g, \mathcal{A}^b$ assigned to each category. To better satisfy the *maximal distance assumption*, we will try different ways to construct these sequences, *i.e.*., assigning colors to each category.

***(i)** Arithmetic sequence on R/G/B channels*: Designing three arithmetic sequences $\mathcal{A}^r, \mathcal{A}^g, \mathcal{A}^b$ for R/G/B channels

Colorization technique	mIoU	aAcc
Arithmetic sequence	85.99	94.37
+ Misalignment start points	86.12	94.45
+ Random additive factors	87.42	95.08
+ Category-specific refinement	**87.73**	**95.29**

Table S.2. **Ablation on *maximal distance assumption*:** The `maskige` reconstruction performance (mIoU and aAcc) of GSS-FF on ADE20K `val` split under different Mask-to-`maskige` transformations $\mathcal{X}$.

respectively. Then we have

$$\mathcal{A}^m = \{a_1^m, a_2^m, \ldots, a_i^m, \ldots, a_n^m\}, m \in \{r, g, b\}. \quad \text{(S.9)}$$

For the i-th color value,

$$a_i^m = a_1^m + (i-1) \cdot k^m, k^m \in N^+, \quad \text{(S.10)}$$

where color channel $m \in \{r, g, b\}$, the interval of arithmetic sequence k^m can be difference between channels, a_1^m default is 0. The set of colors is the Cartesian product of these three series,

$$\mathcal{C} = \mathcal{A}^r \times \mathcal{A}^g \times \mathcal{A}^b. \quad \text{(S.11)}$$

E.g., if the interval of R, G, B channel $k = 45$, the color set $\mathcal{C}$ will be $\{(0,0,0), (0,0,45), \ldots, (225,225,225)\}$.

(ii) *Misalignment start points*: The original starting point of the arithmetic sequence is 0, 0, 0 for R/G/B respectively. In order to avoid duplication of values, we let R/G/B have different starting points,

$$a_1^r \neq a_1^g \neq a_1^b. \quad \text{(S.12)}$$

In practice, we simply set to $a_1^r = 0, a_1^g = 1, a_1^b = 2$.

(iii) *Random additive factors*: Adding three independent random factors $t \in [0, T]$ on the R/G/B arithmetic sequence respectively, to avoid repetition of several same values,

$$a_i^m = a_1^m + (i-1) \cdot k^m + t_i^m. \quad \text{(S.13)}$$

E.g., a color sequence with random additive factors: $\{(1,7,3), (4,2,45), \ldots, (235,215,232)\}$. In practical terms, T is set to 15.

(iv) *Category-specific refinement*: We equip the lower IoU categories with values where the R/G/B values vary at large degrees (*e.g.*, we replace $(128,128,128)$ with $(0,128,255)$). In addition, we keep the color away from gray as possible, because gray is located in the center of the color space, thus being close to many categories and giving rise to a harder learning problem. Such an category-specific refinement allows each category to be possibly furthest from the others as possible.

Results As shown in Table S.2, it is evident that the colorization design for `maskige` generation presents a good amount of impact on the reconstruction performance. In particular, the last design category-specific refinement yields the best results, conforming our intuition and design consideration.

Visualization For visual understanding, in Figure S.7 and Figure S.8 we visualize the 150 colors corresponding to all the categories of ADE20K [23] generated by the *maximal distance assumption* (hand-designed) and gradient descent optimization (learned), respectively. We observe that the hand-designed method produces the colors with enhanced contrast and greater vibrancy. Instead, the colors learned are vibrant for the more frequent categories and relatively dark for the less frequent categories.

C. Overall architecture

Following [6, 16], the modeling of latent prior learning is formulated by an encoder-decoder architecture (See Figure S.2). For the image encoder $\mathcal{I}_\psi$, we take the advantage of hierarchical shifted window transformer [12] for extracting the multi-scale information [19] and memory efficiency [13]. This is different from UViM [10], which uses a single-scale and full-range Transformer as the encoder. To implement the image encoder, we use the Swin-Large architecture [12], pre-trained on ImageNet-22K [5], as the backbone. As shown in Figure S.2, we use four-scale feature maps ($1/4$, $1/8$, $1/16$, $1/32$) and upsample all the lower-resolution features to $1/4$ scale, then concatenate four features across the channel [20]. The multi-level aggregation consists of an MLP and D layers of hierarchical shifted-window Transformer [12], with the swin window size set to 7, the number of attention heads to 16, the embedding dimension to 512, and the FFN dimension to 1024. For the implementation version with resnet as the backbone, $D = 6$. However, for models with strong Swin Transformer backbones, fewer MLA layers are needed, and thus $D = 2$. We implement the `maskige` decoder $\mathcal{D}_\theta$ as a fixed VQVAE decoder [18].

D. More training details

(i) *Latent posterior learning*: As illustrated before, the latent posterior learning is simplified as:

$$\min_{\mathcal{X}^{-1}} \mathbb{E}_{q_{\hat{\phi}}(\hat{z}|\mathcal{X}(c))} \|\mathcal{X}^{-1}(\hat{x}^{(c)}) - c\|. \quad \text{(S.14)}$$

The target can be interpreted as minimizing the distance between a ground-truth segmentation mask and the predicted mask. Following [2, 10], we use cross-entropy loss instead of euclidean distance for a better minimization between segmentation masks.

(ii) *Latent posterior learning for* $\mathcal{X}$: For GSS variants whose $\mathcal{X}$ dose not require training, such as GSS-FF&FT&FT-W, we assign a 3-channel encoding to each category directly based on the *maximum distance assumption*. For the GSS variants that require training, including GSS-TF&TF&, we freeze the parameters of the VQVAE of the DALL-E pretrain and train $\mathcal{X}$ for 4,000 iterations using `SGD` optimizer with a batch size of 16. By the way, the training process of GSS-TT also optimizes the $\mathcal{X}^{-1}$ function.

(iii) *Latent posterior learning for* $\mathcal{X}^{-1}$: We propose a method for training an $\mathcal{X}^{-1}$ that is more robust to noise (used in GSS-FT&FT-W). We found that training $\mathcal{X}^{-1}$ with *noisy* `maskige` helps it learn to be more robust. To provide noisy `maskige`, we can use the trained $\mathcal{I}_\psi$, DALL·E pre-trained $\mathcal{D}_\theta$, and $\mathcal{X}^{-1}$ to directly predict noisy `maskige` predictions. In practice, we use $\mathcal{I}_\psi$ that trained up to the middle checkpoint (*e.g.*, 32,000 iterations) or final checkpoint in latent prior learning. $\mathcal{X}^{-1}$ is trained using cross-entropy loss and optimized with `AdamW`, with a batch size of 16. We trained GSS-FT-W for 40,000 iterations, while GSS-FT was trained for only 3,000 iterations due to its fast convergence. The non-linear function $\mathcal{X}^{-1}$ is implemented using either a convolutional or Swin block structure. Specifically, for GSS-FT, the structure comprises two conv 1×1 layers enclosing a conv 3×3 layer. However, this approach is superseded by GSS-FT-W, the final model, which employs a group of Swin blocks with a number of heads of 4, a Swin window size of 7, and an embedding channel of 128 to realize $\mathcal{X}^{-1}$. Regardless of the specific implementation, $\mathcal{X}^{-1}$ relies on local RGB information in the predicted mask to deduce the category of each pixel.

(iv) *Latent prior learning*: For the optimization of $\mathcal{I}_\psi$, we use the AdamW optimizer and implement a polynomial learning rate decay schedule [22] with a minimum learning rate of 0.0. We set the initial learning rate to 1.5×10^{-3} for Cityscapes and 1.2×10^{-4} for ADE20K and Mseg.

E. Domain generic `maskige` and image encoder

We did two tests by deriving a ***general maskige*** on MSeg [11].

(i) As shown in Table 7 in main paper, we applied our general `maskige` to the Cityscapes dataset and achieved a mIoU score of 79.5, which is only slightly lower than the mIoU score of 80.5 obtained using the Cityspaces specific `maskige`. This result demonstrates the versatility of our `maskige` across different datasets.

(ii) To further evaluate the effectiveness of our domain-generic approach, we shared the image encoder $\mathcal{I}_\psi$ between MSeg and Cityscapes and trained our model on the training split of MSeg. We then evaluated the model on the ***zero-shot*** test split consisting of 6 unseen datasets. As shown in Table 6 in main paper, our GSS outperforms other state-of-the-art methods on the MSeg dataset. These experiments demonstrate that our `maskige` is domain-generic and has the potential for open-world settings.

F. Additional quantitative results

We additionally compare the improved versions of MSeg [11] with 1,500k longer training on the cross-domain benchmark. As shown in Table S.1, despite using short training, our model still achieves better performance. This verifies the advantage of our method in terms of training efficiency, in addition to the accuracy.

G. Additional qualitative results

For further qualitative evaluation, we visualize the prediction results of our GSS on both single-domain segmentation datasets [3, 23] and cross-domain segmentation dataset [11].

As shown in Figure S.3, our GSS has an accurate perception of buses, trucks and pedestrians in distance, whilst also splitting the dense and slim poles. In Figure S.4, we see that GSS correctly recognises a wide range of furniture such as curtains, cabinets, murals, doors and toilets; This suggests that our `maskige` generative approach can accurately represent a wide range of semantic entities. Figure S.5 and Figure S.6 show the cross-domain segmentation performance on images from previously unseen domains (Mseg `test` datasets). It can be seen that GSS performs well in all five datasets in the MSeg `test` split [11], further validating that our generative algorithm has strong cross-domain generalization capabilities.

H. Reproduced semantic segmentation version of UViM [10]

We reproduce UViM with *mmsegmentation* and follow the hyperparameter and structure in the paper [10]. To achieve a fair comparison with our approach, we have made some modifications: **(i)** we implement Swin-Large [12] pre-trained on ImageNet 22K [5] as the Language model LM as ours; **(ii)** we generate the Guiding code straightforwardly in a non-autoregressive manner; **(iii)** we trained 80k iterations in the first stage of UViM [10] and 160k iterations in the second stage. These modifications are necessary to ensure **a fair comparison**.

I. Societal impact

Given that the strong cross-domain generalization capability, we consider our model has the potential to be used in a wide range of visual scenarios. This is desired in practical applications due to the benefits of reducing the demands of per-domain model training and easier deployment and system management. This is meaningful and advantageous in both economics and environment. Conversely, our algorithm may be susceptible to misuse and unintended negative consequences. Thus, it is essential to enforce regulations and oversight for algorithmic applications, ensuring their safe, responsible use for the betterment of humanity and society.

J. Limitations and future work

While our study represents a significant step forward for generative segmentation, our models still fall short of the

performance achieved by top discriminative models. One contributing factor is that decision boundaries for generative models are often less precise than those of discriminative models, resulting in less accurate object edges in segmentation. Another drawback is that generative models require larger amounts of data to achieve good performance, because discriminative models only learn decision boundaries, while generative models need to learn the distribution of the entire sample space. In our experiments, the performance of MSeg is better compared to Cityscapes and ADE20K, which roughly indicates this point.

Additionally, since we convert all categories to colors, the color space is limited, and as the number of categories increases, the colors become more crowded. This can lead to confusion when using $\mathcal{X}^{-1}$ to query and predict the closest pre-defined color for each category from `maskige`, especially near object edges. Therefore, it is worth trying to expand this space to higher dimensions.

Looking ahead, there are several avenues for future research in generative semantic segmentation. One promising direction is instance-level segmentation, which would enable more precise identification and separation of individual objects within an image. Additionally, we believe that it would be valuable to explore a unified model that can perform multiple vision tasks, such as segmentation, 2D object detection, depth prediction, 3D detection, and more.

Given that the second stage training of GSS focuses on latent prior learning, new vision tasks could be inclusively added by incorporating a new posterior distribution of latent variables, without requiring any changes to the model architecture. By pursuing these directions, we believe that significant advances can be made in the field of generative semantic segmentation.

References

[1] Gabriel J Brostow, Julien Fauqueur, and Roberto Cipolla. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognition Letters*, 2009. 3, 9

[2] Ting Chen, Lala Li, Saurabh Saxena, Geoffrey Hinton, and David J Fleet. A generalist framework for panoptic segmentation of images and videos. *arXiv preprint*, 2022. 4

[3] 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*, 2016. 5, 7

[4] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *CVPR*, 2017. 3, 9

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 4, 5

[6] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *CVPR*, 2021. 4

[7] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 2010. 3, 9

[8] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 2013. 3, 9

[9] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint*, 2013. 1

[10] Alexander Kolesnikov, André Susano Pinto, Lucas Beyer, Xiaohua Zhai, Jeremiah Harmsen, and Neil Houlsby. Uvim: A unified modeling approach for vision with learned guiding codes. *arXiv preprint*, 2022. 4, 5

[11] John Lambert, Zhuang Liu, Ozan Sener, James Hays, and Vladlen Koltun. Mseg: A composite dataset for multi-domain semantic segmentation. In *CVPR*, 2020. 3, 5, 9

[12] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021. 4, 5

[13] Jiachen Lu, Jinghan Yao, Junge Zhang, Xiatian Zhu, Hang Xu, Weiguo Gao, Chunjing Xu, Tao Xiang, and Li Zhang. Soft: softmax-free transformer with linear complexity. In *NeurIPS*, 2021. 4

[14] Saeid Motiian, Marco Piccirilli, Donald A Adjeroh, and Gianfranco Doretto. Unified deep supervised domain adaptation and generalization. In *ICCV*, 2017. 3

[15] 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 semantic segmentation in the wild. In *CVPR*, 2014. 3, 9

[16] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *ICML*, 2021. 2, 4

[17] Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. *NeurIPS*, 2018. 3

[18] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *NeurIPS*, 2017. 2, 4

[19] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *CVPR*, 2021. 4

[20] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. In *NeurIPS*, 2021. 4

[21] Oliver Zendel, Katrin Honauer, Markus Murschitz, Daniel Steininger, and Gustavo Fernandez Dominguez. Wilddash-creating hazard-aware benchmarks. In *ECCV*, 2018. 3, 9

[22] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *CVPR*, 2017. 5

[23] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *IJCV*, 2019. 4, 5, 8, 10, 11

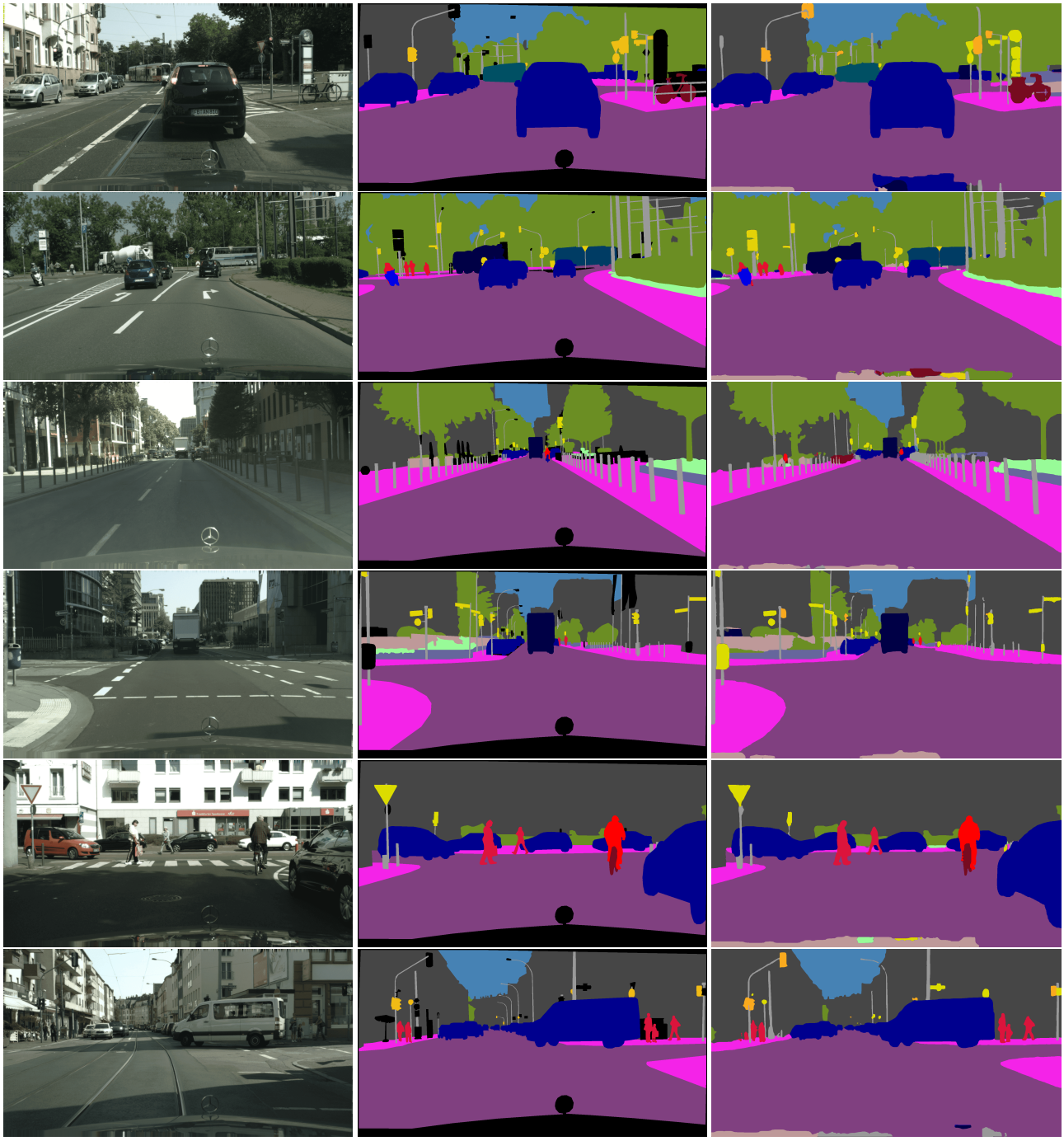

Image Ground Truth Prediction

Figure S.3. Qualitative results of semantic segmentation on Cityscapes `val` split [3].

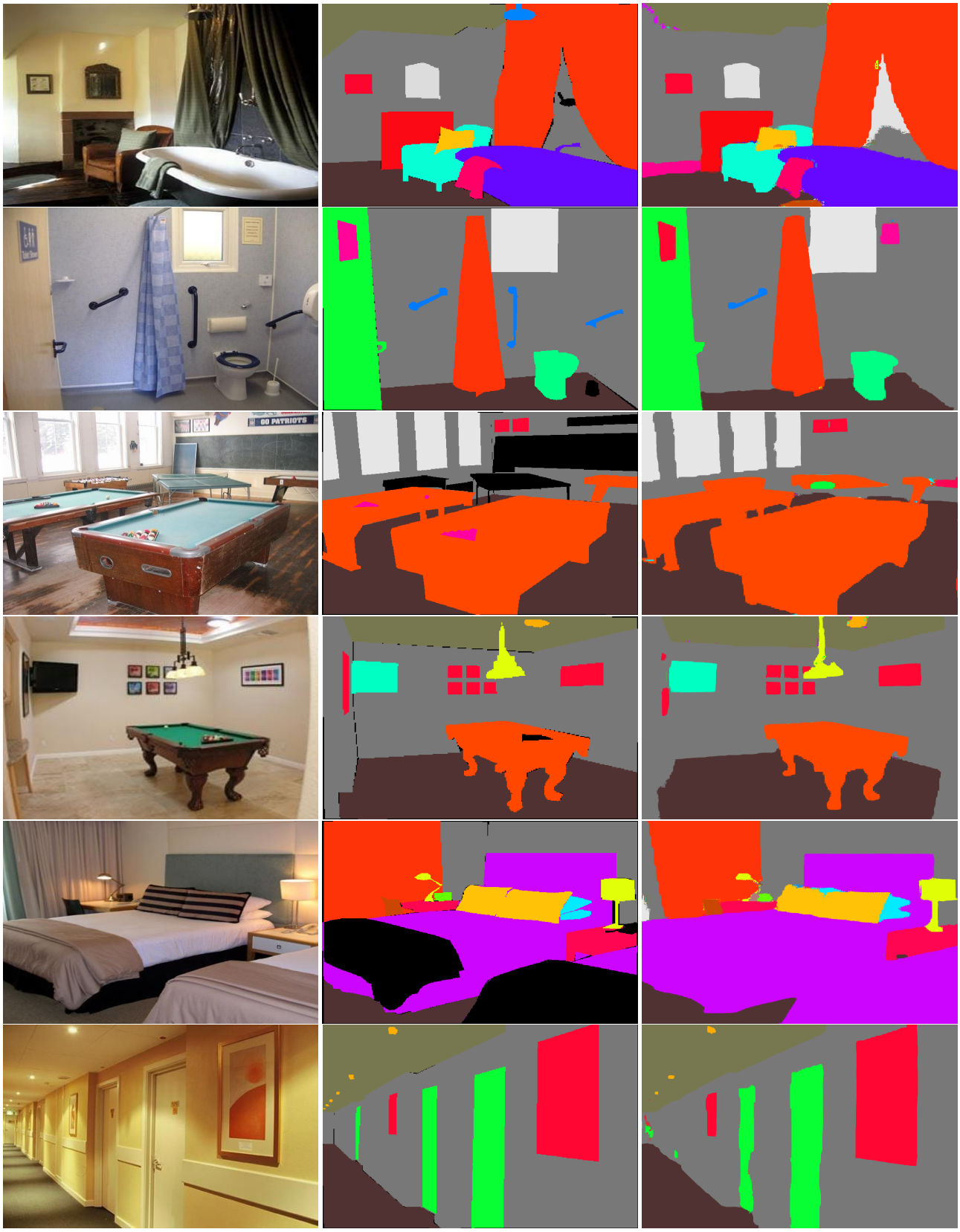

Image Ground Truth Prediction

Figure S.4. Qualitative results of semantic segmentation on ADE20K `val` split [23].

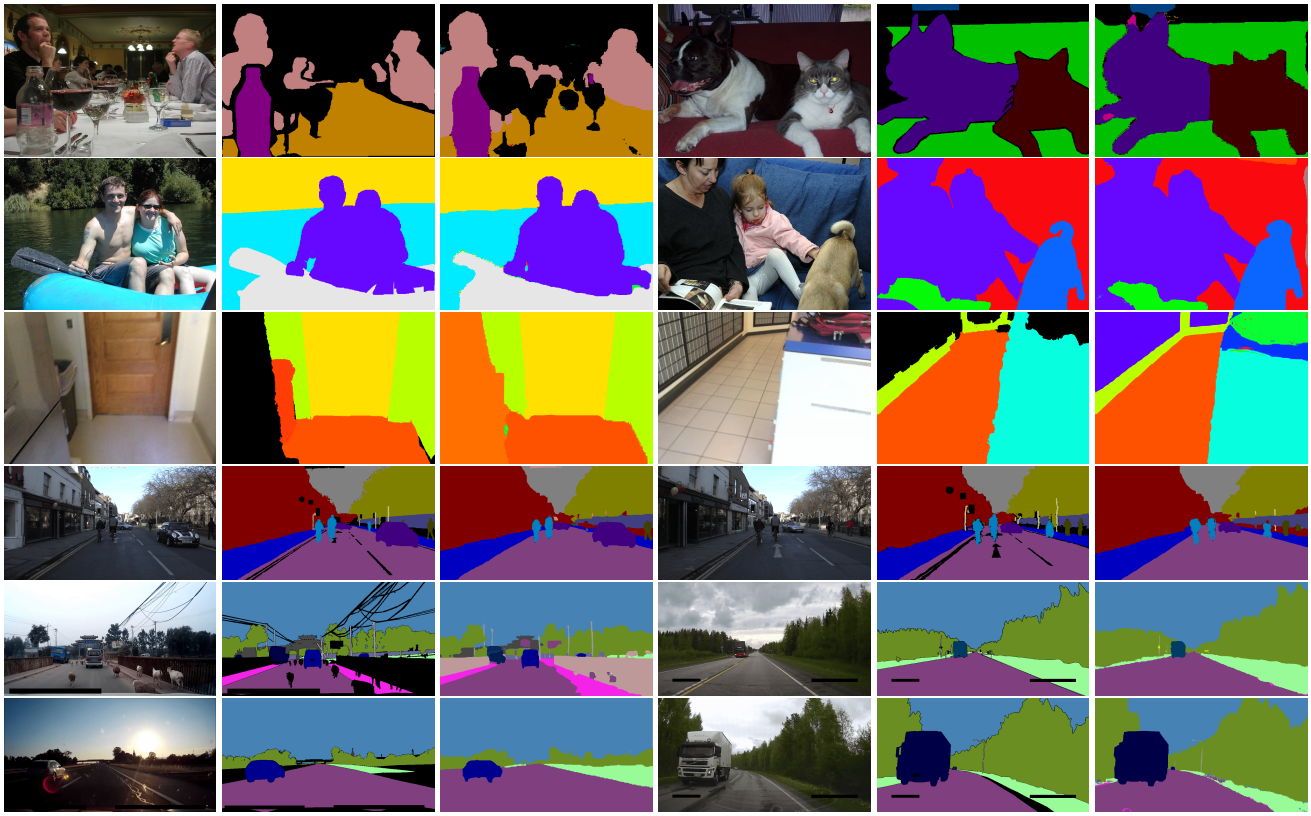

Figure S.5. Qualitative results of semantic segmentation on MSeg `test` datasets [11]. From top to bottom: Pascal VOC [7], Pascal Context [15], ScanNet-20 [4], CamVid [1] and WildDash [21] (the last two rows).

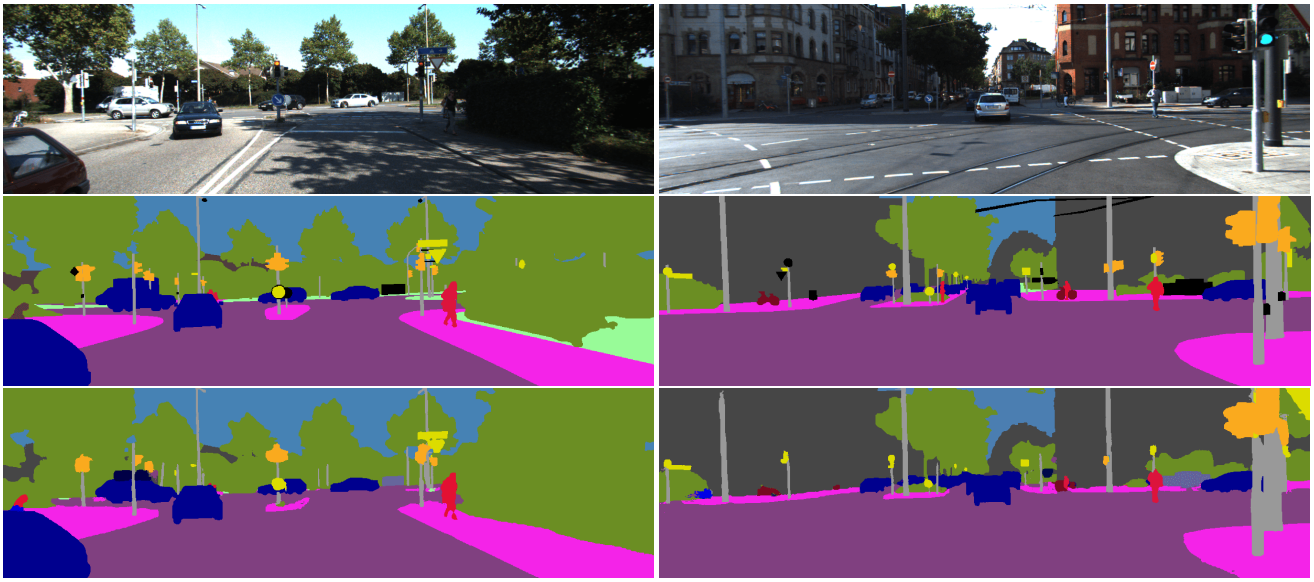

Figure S.6. Qualitative results of semantic segmentation on MSeg `test` dataset [11] (KITTI dataset [8]). The 1^{st} row is input image, the 2^{rd} row is Ground Truth, and the 3^{rd} row is prediction result.

Figure S.7. Visualization of `maskige` for each category in ADE20K [23] dataset under *maximal distance assumption*.

Figure S.8. Visualization of `maskige` for each category in ADE20K [23] dataset under gradient descent optimization.