Beyond Text: Visual Description Assembly by Probabilistic Model for CLIP-based Weakly Supervised Semantic Segmentation–Appendix

1. More Implementation Details

Following [6, 7], our adapter architecture first uses 12 MLP layers to separately project the features from each layer of the ViT-B CLIP Image Encoder. Subsequently, the 12 projected features are concatenated and compressed to 256 channels via a convolutional layer. The decoder consists of three transformer layers, each with an output dimension of 256. For data augmentation, training images are processed with random horizontal flipping, random scaling with ratio [0.5, 2.0], and random cropping to 320 × 320. Our training strategy is staged: we first warm-up the segmentation network using only the template text for 8000 iterations on VOC or 20000 iterations on COCO. After this, we begin the INN training by first warming up the inter-class GMM for 2000 iterations on VOC or 8000 iterations on COCO, and then train the inter-class and intra-class GMMs simultaneously. During inference, the final decoder prediction masks are refined using multi-scale and DenseCRF [3] post-processing techniques.

2. Details about Invertible Neural Network

Following Real-NVP [1], the invertible neural network (INN) we use consists of N invertible layers with the same structure. In detail, for each invertible layer the input feature u is first equally split into u_1 and u_2 at the channel level. Then the latent features $z_2 = u_2 \odot \exp\left(s\left(u_1\right)\right) + t\left(u_1\right)$ and $z_1 = u_1$, where s and t are two learnable neural networks as in [1]. Then z_1 and z_2 are concatenated at the channel level to z. Finally, z is send to a ActNorm Layer as in [2]. The number of invertible layer in our work is set to 6.

3. INN Loss Function Derivation

For the derivation from Eq.(4) to Eq.(5) in main paper, the distribution of GMM can be first represented as:

$$p_Z(z) = \sum_{k=1}^{K} \pi_k \cdot \mathcal{N}(z|\mu_k, \Sigma_k).$$

We first assume the Σ_k is identity matrix $\mathbb{I}$, then the Probability Density Function (PDF) of the k-th inter-class GMM component can be represented as:

$$\begin{aligned}\mathcal{N}(z|\mu_k, \mathbb{I}) &= \frac{1}{\sqrt{(2\pi)^D|\mathbb{I}|}} \exp\left(-\frac{1}{2}(z-\mu_k)^T\mathbb{I}^{-1}(z-\mu_k)\right) \\ &= \frac{1}{(2\pi)^{D/2}} \exp\left(-\frac{1}{2}||z-\mu_k||_2^2\right).\end{aligned}$$

According to our definition of the learnable log weights $c_k = log(\pi_k)$ which equal to $\pi_k = \exp(c_k)$, $p_Z(z)$ can be converted as follow:

$$p_Z(z) = \sum_{k=1}^{K} \exp(c_k) \cdot \left[\frac{1}{(2\pi)^{D/2}} \cdot exp\left(-\frac{1}{2}||z-\mu_k||_2^2\right)\right].$$

Since we have defined the negative log likelihood of the k-th component as $E_k(z, \mu_k) = \frac{1}{2}||z-\mu_k||_2^2$, so that we simplify the $-log\ p_Z(z)$ as follow:

$$-log\ p_Z(z) = -log\left[\frac{1}{(2\pi)^{D/2}} \sum_{k=1}^{K} exp\left(c_k - E_k(z, \mu_k)\right)\right].$$

Substituting the above $-log\ p_Z(z)$ into Eq.(5) and ignore the constant $log((2\pi)^{D/2})$ which has none influence on the loss optimization, the $\mathcal{L}_{nll}$ can be represented as:

$$\mathcal{L}_{nll} = \mathbb{E}\left[-log\left(\sum_{k=1}^{K} exp\left(c_k - E_k(z, \mu_k)\right)\right) - log|det\ J|\right].$$

We define the $\mathrm{LSE}(\cdot)$ operation as:

$$LSE(v_1, ..., v_K) = log\left(\sum_{k=1}^{K} exp(v_k)\right).$$

Then substituting it into above $\mathcal{L}_{nll}$, the final $\mathcal{L}_{inter}$ can be represented as:

$$\mathcal{L}_{inter} = \mathbb{E}[-\underset{k}{\mathrm{LSE}}(c_k - E_k(f_\theta(x), \mu_k)) - log|det\ J|].$$

For the intra-class loss $\mathcal{L}_{intra}$, it follows a similar above derivation process.

4. Attribute Response Analysis

To verify that our intra-class GMM indeed captures the diverse attributes within class, we visualize the CAMs activated by the 5 intra-class GMM components with strongest attribute responses in Fig. 1. It can be clearly observed that different components activate distinct parts of the target class, such as the head of dog, the tail of airplane, the wheels of bus, and so on.

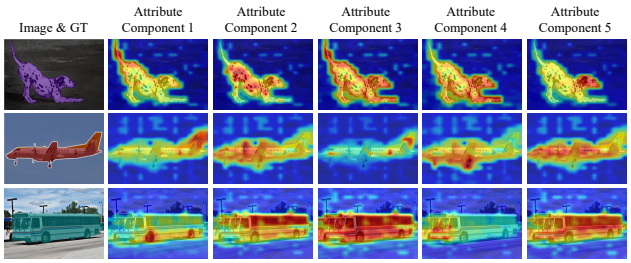

Figure 1. The CAMs visualization results activated by 5 attribute components in our intra-class GMM with the strongest responses.

Table 1. Training efficiency comparisons on VOC dataset.

Method	Traing Time	GPU memory	mIoU(%)
$\text{ToCo}_{\text{cvpr23}}$ [4]	506 mins	17.9 G	71.1
$\text{SeCo}_{\text{cvpr24}}$ [5]	407 mins	17.6 G	74.0
$\text{WeCLIP}_{\text{cvpr24}}$ [7]	270 mins	6.2 G	76.4
$\text{ExCEL}_{\text{cvpr25}}$ [6]	90 mins	3.2 G	78.4
Ours	117 mins	4.5 G	79.9

5. Efficiency Comparisons

Tab. 1 presents the comparison of training efficiency between our method and several previous single-stage WSSS methods. It can be found that our method is significantly superior to ToCo, SeCo, and WeCLIP in terms of training time, GPU memory, and mIoU performance. Since our method requires training an additional INN, it incurs increased training time and GPU consumption compared to ExCEL. Nevertheless, the 1.5% mIoU performance improvement brought by this extra overhead is worthwhile. It must be pointed out that the high efficiency of ExCEL comes at a cost. It relies heavily on large language models (LLMs) to generate rich text descriptions, which introduces complex external dependencies, and its training time (90 min) does not include the time required for LLM debugging. In contrast, our method directly mines visual attribute information from CLIP and dynamically assembles it into visual descriptions without the need for LLMs. Therefore, our method is still very competitive regarding the overall training efficiency. For the inference latency, the INN-related components are only used to generate better CAM as dense supervision during training and removed in inference phase. Thus our method retains exactly same network structure as ExCEL in inference, ensuring identical inference latency with it.

6. Upper bound and future direction

Upper bound of our current implementation is likely influenced by structure of latent space. While the standard GMM in Euclidean latent space is robust and effective for current WSSS benchmarks, our method is extensible to meet more task demands. For instance, in complex fine-grained recognition task involving highly similar visual attributes (e.g., crow vs. sparrow), the non-Euclidean hyperbolic space GMM presents a promising avenue. Its negative curvature geometry theoretically facilitates distinct separation of subtle attributes with enough distance than flat Euclidean spaces. Exploring such diverse latent space formulations to adapt broader tasks will be our future focus. Besides, the modular nature of our approach ensures its scalability to different architectures and larger datasets.

References

[1] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016. 1

[2] Durk P Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *Advances in Neural Information Processing Systems*, 31, 2018. 1

[3] Philipp Krähenbühl and Vladlen Koltun. Parameter learning and convergent inference for dense random fields. In *International conference on machine learning*, pages 513–521. PMLR, 2013. 1

[4] Lixiang Ru, Heliang Zheng, Yibing Zhan, and Bo Du. Token contrast for weakly-supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3093–3102, 2023. 2

[5] Zhiwei Yang, Kexue Fu, Minghong Duan, Linhao Qu, Shuo Wang, and Zhijian Song. Separate and conquer: Decoupling co-occurrence via decomposition and representation for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3606–3615, 2024. 2

[6] Zhiwei Yang, Yucong Meng, Kexue Fu, Feilong Tang, Shuo Wang, and Zhijian Song. Exploring clip's dense knowledge for weakly supervised semantic segmentation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 20223–20232, 2025. 1, 2

[7] Bingfeng Zhang, Siyue Yu, Yunchao Wei, Yao Zhao, and Jimin Xiao. Frozen clip: A strong backbone for weakly supervised semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3796–3806, 2024. 1, 2