Synthesizing Image with High-Quality Segmentation Mask by Prompting Large Vision Model

Tuyen Tran
Applied AI Institute, Deakin University, Australia
t.tran@deakin.edu.au

Abstract

This paper presents a novel pipeline to generate high-quality segmentation masks for synthetic datasets produced by a text-to-image generative model. In contrast to previous approaches that directly apply a threshold on the attention map extracted during generation process, we leverage this map to prompt a large vision model. We extract a set of candidate point prompts from the attention maps and then select a subset that maximizes diversity within the candidate set. These selected points prompt the vision model, yielding fine-grained segmentation masks. To validate our method, we trained segmentation models on synthetic datasets and evaluated them on real datasets, including PASCAL VOC and MSCOCO. Both qualitative and quantitative results demonstrate the superior quality of segmentation masks produced by our method compared to other thresholding baseline approaches.

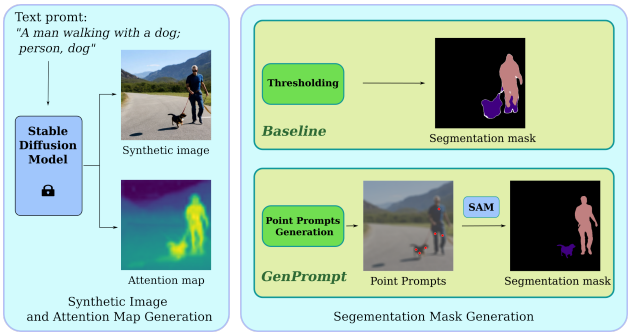


Figure 1. Prior approaches applying a threshold on the attention map yield coarse, low-quality segmentation masks. In contrast, we suggests using the attention map to generate prompts for a promptable vision model (e.g., SAM). To this end, we propose the GenPrompt algorithm, which generates points (marked in red circles) for prompting the segmentation model, resulting in high-quality masks.

1. Introduction

The development of generative models like Stable Diffusion (SD) has unlocked the potential of using synthetic data to train Artificial Intelligence (AI) systems, offering advantages in scalability and accessibility. While SD model can effectively produce photorealistic dataset with great diversity, it remain challenging to get high quality segmentation mask, where we need the annotation at pixel level. Simply applying a pre-trained segmentation model to synthetic images is not a valid option, because the goal is to generate a synthetic dataset across various domains, potentially beyond the seen categories of the pre-trained model[1].

One pioneering research is Diffumask [3], where they demonstrate the potential of extracting text-guided cross-attention information in SD model for localizing class-specific regions within synthetic images. To convert the resulting cross-attention maps into usable annotation masks, they estimate an adaptive threshold for each class present in the synthetic image. Another work, Dataset Diffusion [2] proposed to refine the cross-attention map by multiplying it with a power of the self-attention map. They then empirically select a fixed threshold to transform the refined attention map into a segmentation label mask. While Diffumask and Dataset Diffusion illustrate the feasibility of generating semantic masks through attention maps, producing high-quality segmentation masks remains challenging due to the difficulty in determining the optimal threshold for each specific case.

To address this challenge, we propose a different strategy of prompting a large vision model. Instead of applying a threshold to the cross-attention map, we use this map to generate point prompts for a promptable vision model. This approach, requiring only a small number of high-certainty prompt points, significantly improves segmentation performance. Additionally, the promptable model can generate segmentation masks for arbitrary image areas based on given prompts, without relying on specific categories. This fulfills

[1]Synthetic data generation also holds promise for tackling tasks with objectless regions of interest, like satellite imagery segmentation or partial object segmentation, as explored in [2].

the task's requirements for domain-agnostic image generation. We specifically employ the Segment Anything Model[2] (SAM), known as one of the most effective promptable models for segmentation task.

The remaining challenge is to generate a suitable set of point prompts for producing high-quality segmentation masks. We explored various approaches and found that maximizing the distance between points is the most effective strategy. Figure 1 illustrates an overview of our approach. In contrast to the thresholding methods described in [2, 3], we utilize the attention map to generate point prompts for SAM. Specifically, we propose GenPrompt, an algorithm that generates point prompts with maximum diversity. The diversity score we use is the mean harmonic distance between points, which encourages the selected points to be far apart and evenly distributed over the target area. To evaluate the effectiveness of our proposed method, we train two deep learning models, DeepLabV3 and Mask2Former on the generated dataset to and evaluate them on the validation sets of PASCAL VOC and COCO. Both quantitative and qualitative results indicate the effectiveness of our approach compared to the thresholding baselines.

2. Methodology

2.1. Attention map generation

The pipeline overview is illustrated in Figure 2. Following the procedures described in [2, 3], we harness the SD model to generate images and their attention maps (probability maps) from textual input. First, text encoder in SD encodes the text prompt to embedding $e \in \mathbb{R}^{\ell \times d_e}$, where ℓ represents the text length and and d_e denotes its dimension. We use the same text prompt $\tilde{S} = [S; C]$ as in [2], where S is an image caption and $C = [\mathsf{c}_1; \mathsf{c}_2; ...; \mathsf{c}_M]$ represents M class objects in an image. The embedding e guides the denoising process over T step of the SD. During these steps, the SD transform the initial latent state $z_T = \mathcal{N}(0, 1)$ into the final latent state z_0 residing in $\mathbb{R}^{H \times W \times d}$, where H, W and d represent size of z_0. In each step t, the transformation of z_t to z_{t-1} occurs across L layers of self- and cross-attention within the UNet framework. For each layer ℓ and time step t, we extract the self-attention map:

$$A_S^{\ell,t} = \text{Softmax}\left(\frac{Q_z K_z^\top}{\sqrt{d_\ell}}\right) \in [0,1]^{HW \times HW}, \quad (1)$$

where Q_z, K_z are query, key of z_t obtained from linear transformation, and d_ℓ is the feature length at layer ℓ. Intuitively, the extracted self-attention maps illustrate the pairwise correlations among each positions within the latent variable z_t. Similarly, the cross-attention maps represent how each specific location in the image space correlates with each word

[2]https://github.com/facebookresearch/segment-anything.

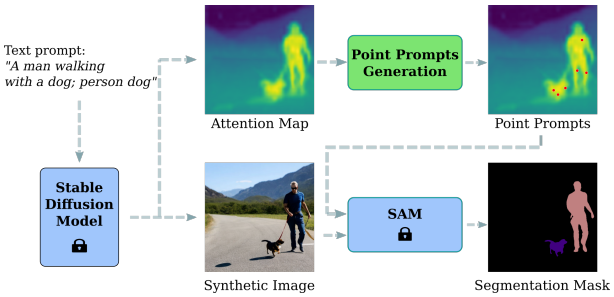


Figure 2. **Pipeline for the Synthetic Image and Segmentation Mask Generation:** In the first stage, we use the SD model to generate a synthetic image and attention map based on a given text prompt (Section 2.1). The attention map is then used in the second stage for prompt generation. Here, we introduce an algorithm to generate point prompts (marked in red circles) that maximize diversity within the candidate set (Section 2.2). These points are used to prompt the SAM model, producing the final segmentation mask.

token of the text embedding, which could be expressed as:

$$A_C^{\ell,t} = \text{Softmax}\left(\frac{Q_z K_e^\top}{\sqrt{d_l}}\right) \in [0,1]^{HW \times M}. \quad (2)$$

The aggregated self-attention and cross-attention maps are derived by averaging all maps over layers and timesteps:

$$A_S = \frac{1}{L \cdot T} \sum_{(\ell=1,t=1)}^{(L,T)} A_S^{\ell,t}, \; A_C = \frac{1}{L \cdot T} \sum_{(\ell=1,t=1)}^{(L,T)} A_C^{\ell,t}. \quad (3)$$

Finally, we follow [2] to obtain the probability map P by exponentiating the attention map A_S to the power of r before multiplying it by with A_C :

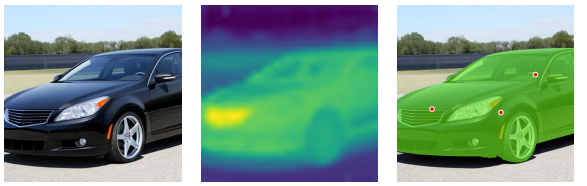
$$P = (A_S)^r \cdot A_C \in [0,1]^{HW \times M}. \quad (4)$$

The probability map P can be equivalently presented as set $\{p_{ij}^m\}$, where i from 1 to W, j from 1 to H and m from 1 to M. Each element p_{ij}^m represents *the confidence score that the pixel at position* (i, j) *in the generated image belongs to the class* m. However, the probability map P is still coarse-grained, so we propose to use it only for generating point prompts in the next step.

2.2. GenPrompt: Generating Point Prompts with Maximum Diversity

Objective of the GenPrompt Algorithm

Prompt selection strategy greatly effects on the segmentation quality, which is illustrated in Figure 3. Using a single point with the highest certainty can lead to ambiguities for the SAM model. Selecting multiple points with the highest certainty does not improve much, as these points are likely close to each other, making it difficult for SAM to segment



(a) Synthetic Image (b) Attention Map (c) Maximize diversity

(d) Single prompt (e) Multiple prompts (f) Random prompts

Figure 3. Comparing segmentation results of different strategies to generate point prompts from an input attention map.

the entire object. Randomly selecting a set of points can result in selecting points that are too close together or have low certainty, which can damage the segmentation quality. Based on these observations, we suggest two criteria for a good set of point prompts. First, all points in the set should have high certainty to ensure they belong precisely to the target object. Second, these points should be sufficiently far apart to evenly distribute across the entire object. With this intuitive, we formulate the point prompts generation problem as the Maximum Diversity Problem (MDP)[1], aiming to select a subset of elements from a larger set to maximize the diversity score among them. We define the diversity score as the harmonic mean of the distances between points, encouraging even distribution over the object and preventing points from being too close together. Formally, let M represent the number of class objects in the image, and P be the certainty map derived from Equation 4. Since a pixel location is exclusive to a single object, we obtain initial coarse probability map for class m by executing argmax operation along the class dimension of P:

$$\hat{p}_{ij}^{m} = \begin{cases} p_{ij}^{m} & \text{if } m = \arg\max_{\eta}(p_{ij}^{\eta}) \\ 0 & \text{otherwise} \end{cases}. \tag{5}$$

We define $\mathcal{C}^m$ as a set of candidate point (i, j) taken from P such as for each point $(i, j) \in \mathcal{C}^m$, the associated element $\hat{p}_{ij}^{m} > 0$:

$$\mathcal{C}^m = \left\{(i,j) \mid \hat{p}_{ij}^{m} > 0 \text{ , and } 1 \le i \le W, 1 \le j \le H\right\}. \tag{6}$$

For each category m, our objective is to find a subset $\mathcal{S}^m = \{(i_1, j_1), (i_2, j_2), \ldots, (i_K, j_K)\}$ containing K points selected from $\mathcal{C}^m$ to maximize $H(\mathcal{S}^m)$ as the harmonic mean of distance among the points in S^m. The optimization problem could be defined as:

$$\max_{\mathcal{S}^m} H(\mathcal{S}^m) = \left(\frac{1}{\Omega} \sum_{a=1}^{K-1} \sum_{b=a+1}^{K} \frac{1}{d\left((i_a, j_a), (i_b, j_b)\right)} \right)^{-1}, \tag{7}$$

subject to:

$$\mid \mathcal{S}^m \mid = K \text{ and } \mathcal{S}^m \subseteq \mathcal{C}^m.$$

In Equation 7, $d\left((i_a, j_a), (i_b, j_b)\right)$ is the Euclidean distance between points (i_a, j_a) and (i_b, j_b), and is given by:

$$d\left((i_a, j_a), (i_b, j_b)\right) = \sqrt{(i_a - i_b)^2 + (j_a - j_b)^2}, \tag{8}$$

and Ω is defined as the total number of unique pairs in $\mathcal{S}^m$, which is equal to $\frac{K(K-1)}{2}$.

Procedure of the GenPrompt Algorithm

The MDP is NP-hard, implying that there is no known polynomial-time algorithm for its efficient solution [1]. However, an optimal solution is not necessary because a set of high-certainty points reasonably far apart can already yield high-quality segmentation results with the SAM model. Therefore, we use a Random Sampling approach with uncertainty awareness to find a satisfactory solution. Formally, we firstly initialize the max diversity score $D_{\text{max}} = 0$ and the selected subset $\mathcal{S}_{\text{sel}} = \emptyset$ then repeat the iteration N time. For each iteration n, where $n = 1, 2, ..N$, we sample K high-certainty elements to form the subset $\mathcal{S}_{\text{tmp}}$ from the candidate prompt points $\mathcal{C}^m$:

$$\mathcal{S}_{\text{tmp}} = \{(i_k, j_k)\}_{k=1}^{K} \subseteq \{(i_e, j_e) \in \mathcal{C}^m \mid \hat{p}_{i_e j_e} > \alpha\}, \tag{9}$$

where α is a threshold certainty to guarantee the selected point belong to specific target object. Then, the diversity score is calculate using function $H(.)$ defined in Equation 7

$$D_{\text{tmp}} = H(\mathcal{S}_{\text{tmp}}). \tag{10}$$

If D_{tmp} exceeds the previous iteration's D_{max} , we update $D_{\text{max}} = D_{\text{tmp}}$ and $\mathcal{S}_{\text{sel}} = \mathcal{S}_{\text{tmp}}$, accordingly. This iterative process continues until reaching the maximum iteration N. Once the S_{sel} with the highest diversity score is obtained, it is assigned to $\mathcal{S}^m$ as the set of augmentation point prompts for object m. This procedure is repeated for all M objects in the image to derive the final set of augmentation point prompts $\mathcal{S}$. Finally, the set of point prompts $\mathcal{S}$, along with the originally generated image I, are used as inputs to generate the segmentation mask I_{mask} using the SAM.

3. Experiments

3.1. Experimental setup

Baselines: We compare against two recent works: Dataset Diffusion [2] and DiffuMask [3].

Training set	Model	VOC *(mIOU)*	COCO *(mIOU)*
Real Data	DeepLabV3, R50	77.4	48.9
	DeepLabV3, R101	79.9	54.9
	Mask2Former, R50	77.3	57.8
DiffuMask [3]	Mask2Former, R50	57.4	-
Dataset Diffusion [2]	DeepLabV3, R50	61.6	32.4
	DeepLabV3, R101	64.8	34.2
	Mask2Former, R50	60.2	31.0
GenPrompt	DeepLabV3, R50	65.1	35.2
	DeepLabV3, R101	68.3	36.1
	Mask2Former, R50	64.0	33.8

Table 1. Semantic segmentation performance of DeepLabV3 and Mask2Former models trained on different dataset.

Model and dataset: Following [2, 3], we evaluate the generated dataset's quality by training standard segmentation models, DeepLabv3 and Mask2Former on $40k$ images for VOC dataset and $80k$ images for COCO dataset.

3.2. Quantitative results

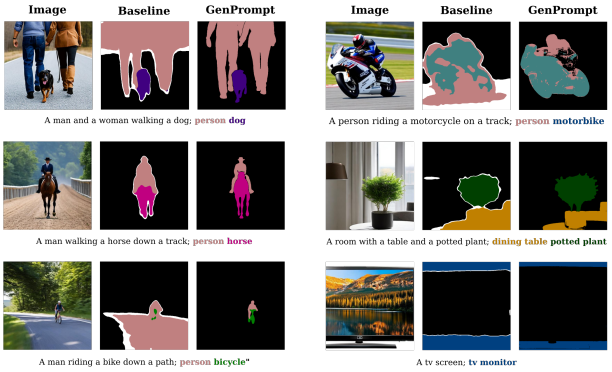
Table 1 compares the semantic segmentation performance of the DeepLabV3 and Mask2Former models when trained on real datasets and synthetic datasets generated through various methods. The GenPrompt approach, which leverages prompting techniques, clearly outperform thresholding approaches, with gains of 6.6 mIOU over DiffuMask and 3.8 mIOU over Dataset Diffusion on the VOC dataset. Although GenPrompt demonstrates encouraging capabilities, a substantial performance gap of over 10 mIOU remains between our synthetic datasets and real datasets. This gap is partly attributed to the limitations of the SD model for generating complex scenes from text prompts, as discussed in [2]. Additionally, the efficacy of our approach relies on the precision of initial attention maps to enable effective prompt selection. Without sufficiently precise attention maps, our approach may struggle to achieve high-quality annotations.

3.3. Qualitative results

Figure 4 displays qualitative results, comparing our method to the baseline Dataset Diffusion [2]. Object masks are color-coded to match the object names in the caption.

Figure 4a presents some successful cases. As shown, our method yields segmentation masks with significantly higher precision compared to baseline outputs. GenPrompt reaches optimal capability in cases where the initial probability maps can reasonably approximate visual characteristics of the intended objects.

We also demonstrate limitations of our method in Figure 4b. When the attention map fails to provide an accurate candidate set, our method may not perform optimally, leading to incorrect data annotations. This impacts the training of the segmentation model, explaining why although our method can produce more precise segmentation masks in many cases, the gain in quantitative results is reasonable and still exits a notable gap compared to training on real data.



(a) Successful Cases. (b) Failure Cases.

Figure 4. **Qualitative analysis of GenPrompt algorithm:** Figure (a) shows successful cases. With sufficiently accurate attention maps, our method could produce fine-grained segmentation annotation (Row 1). Our approach also effectively handles challenging scenarios from the baseline method [2], such as closely intertwined objects (Row 2) or small objects (Row 3). Figure (b) analyzes failure cases, where poor-quality attention maps can result in selecting incorrect prompt points, decreasing segmentation mask quality (Row 1, 2). A notable error occurs in Row 3, where the TV and background are misclassified, leading to an erroneous segmentation mask despite the potential for fine-grained segmentation.

4. Conclusion

In this work, we introduce GenPrompt, an algorithm that utilizes the extracted attention map to generate prompts for synthetic dataset generation in segmentation tasks. By using prompting technique, our work can yields substantially more accurate mask annotations compared to the thresholding baselines. This enhanced performance is accomplished without compromising the critical requirement of generating images in diverse domains. We hope our proposed framework, supported by detailed analysis, will facilitate further exploration in this promising research direction.

References

[1] Ching-Chung Kuo, Fred Glover, and Krishna S Dhir. Analyzing and modeling the maximum diversity problem by zero-one programming. *Decision Sciences*, 24(6):1171–1185, 1993.

[2] Quang Ho Nguyen, Truong Vu, Anh Tran, and Khoi Nguyen. Dataset diffusion: Diffusion-based synthetic dataset generation for pixel-level semantic segmentation. In *Thirty-Seventh Conference on Neural Information Processing Systems*, 2023.

[3] Weijia Wu, Yuzhong Zhao, Mike Zheng Shou, Hong Zhou, and Chunhua Shen. Diffumask: Synthesizing images with pixel-level annotations for semantic segmentation using diffusion models. *Proc. Int. Conf. Computer Vision (ICCV 2023)*, 2023.

Synthesizing Image with High-Quality Segmentation Mask by Prompting Large Vision Model

Supplementary Material

In this supplementary document we include:

- Pseudocode 1 the GenPrompt algorithm described in Section 2.2 of the main paper.
- Additional implementation details.
- Ablation study and additional qualitative results.

5. Implementation details

Training procedure

To evaluate the effectiveness of the synthetic dataset, we assess the performance of existing segmentation models trained on it. As in [2, 3] we ensure a fair comparison by applying the same training procedure used for real data. Formally, given a segmentation model $\mathcal{F}$, the predicted segmentation mask Y for the generated image I is obtained as:

$$Y = \mathcal{F}(I). \tag{11}$$

The loss $\mathcal{L}$ used to update parameters in model $\mathcal{F}$ is calculated via the cross-entropy loss function $\mathcal{L}_{\text{CE}}$:

$$\mathcal{L} = \mathcal{L}_{\text{CE}}(Y, I_{\text{mask}}). \tag{12}$$

Hyperparameter setting

We set the threshold of certainty α to 0.8 for extracting candidate prompts. The number of iteration N is set to 100 to find the optimal point prompt set from the candidate set . For each object of interest in the image, we select 3 point prompts $(K = 3)$. The other parameters used to generate the attention map are kept consistent with those in [2]. We utilized the text prompt provided in [2] to generate synthetic images using Stable Diffusion version 2.1-base [3]. We evaluate the model's performance on the validation sets of the VOC and COCO datasets. The training procedure is based on the MMsegmentation framework [4]. We use the AdamW optimizer with a learning rate of $1e-4$, weight decay of $1e-4$, and train for $20,000$ iterations with a batch size of 8. We follow the standard MMSegmentation framework for the other model hyperparameters. We conducted the training using a server with four Tesla V100 GPUs, each with 32GB of memory, an Intel Xeon E5-2698 processor, and 256GB of RAM. For synthetic image and segmentation mask generation, we ran the process in parallel over eight V100 GPUs, which took ten hours to generate $40,000$ data samples.

[3] https://huggingface.co/stabilityai/stable-diffusion-2-1-base

[4] https://github.com/open-mmlab/mmsegmentation

6. Additional Results

6.1. Ablation study

For all experiments in the ablation study, we use the DeepLabv3 model and evaluate on the VOC dataset, with the same configuration as described in Section 3.1.

Table 2 shows the impact of varying point prompt quantity on the final performance. As has been discussed in Section 2.2, using a small number of point prompts lead to the performance degradation. However, we observed that increasing the number of points beyond a certain threshold did not yield notable performance improvements. We selected three point prompts to achieve a balance between performance and computational cost.

Table 3 showcases the results of employing various point prompt generation techniques. As detailed in Section 2.2 and Figure 3, straightforward strategies exhibit limitations in producing high precise segmentation mask for the whole object. On the other hand, our proposed GenPrompt method, which generates point prompts with maximized diversity, achieves the highest quality segmentation masks from the given probability maps.

We explore the impact of generated image quantity on segmentation model performance in Table 4. We observed there was a positive but diminishing marginal effect of increasing synthetic data quantity on performance. This suggests that the approach may have reached a limitation, where additional data provides minimal benefit. This also aligns with the findings of [2] regarding the effect of synthetic data size.

The effect of threshold α to extract candidate points is present in Table 5. Note that, unlike baseline methods that apply the threshold directly to binarize the attention map, we use the threshold only to extract the candidate point prompts. The purpose of this is to filter out points with low certainty (usually the point at the object boundary), to ensure the the selected point prompts belonging to the target object. We observe that the performance is not much sensitive to the choice of α. However, if choose the threshold too high, candidate prompts will not cover most part of object and will tend to concentrate on local area. The value of alpha as 0.8 achieve the best performance of 65.1 mIOU.

# point prompts	mIoU (%)
2	52.3
3	65.1
4	64.8

Table 2. Impact of different number of point prompts.

Algorithm 1 Generating Point Prompts with Maximum Diversity.

Input:
- Number of object classes M
- Probability map $P = \{p_{ij}^m\}$, where each p_{ij}^m represents the certainty of pixel at location (i, j) belonging to object class m

Output:
- Point prompts set for M object classes $\mathcal{S} = \{\mathcal{S}^m\}_{m=1}^M$

Initialize $\mathcal{S}$ as an empty set for all classes $m \in 1, ..., M$
for $m = 1$ *to* M **do**
  foreach *point* (i, j) **do**
    if $m = \arg\max_\eta(p_{ij}^\eta)$ **then** $\hat{p}_{ij}^m \leftarrow p_{ij}^m$ `// Obtain coarse probability map for class m;`
    else $\hat{p}_{ij}^m, \leftarrow 0$;
  end
  Define $\mathcal{C}^m$ as set $\{(i, j) \mid \hat{p}_{ij}^m > 0\}$ and initialize D_{max} as 0 `// Construct the candidate set`
  for $n = 1$ *to* N **do**
    $\mathcal{S}_{\text{tmp}} \leftarrow$ Randomly select K points that have a certainty above α from $\mathcal{C}^m$
    $D_{\text{tmp}} \leftarrow$ Compute harmonic mean of distances between all pairs of points in $\mathcal{S}_{\text{tmp}}$ `// Refer to Equation 10`
    if $D_{tmp} > D_{max}$ **then** Update D_{max} with D_{tmp} and assign $\mathcal{S}_{\text{tmp}}$ to $\mathcal{S}_{\text{sel}}$;
  end
  Assign set $\mathcal{S}_{\text{sel}}$ to $\mathcal{S}^m$
end
return $\mathcal{S}$

Prompts selection	mIoU (%)
Single point	38.8
Multiple points	43.6
Random	61.0
Maximum diversity	65.1

Table 3. Performance of various point prompts generation methods.

# images	mIoU (%)
20k	64.0
30k	65.0
40k	65.1

Table 4. Impact of different number of generated images.

α	mIoU (%)
0.7	64.8
0.8	65.1
0.9	64.6

Table 5. Analysis of α.

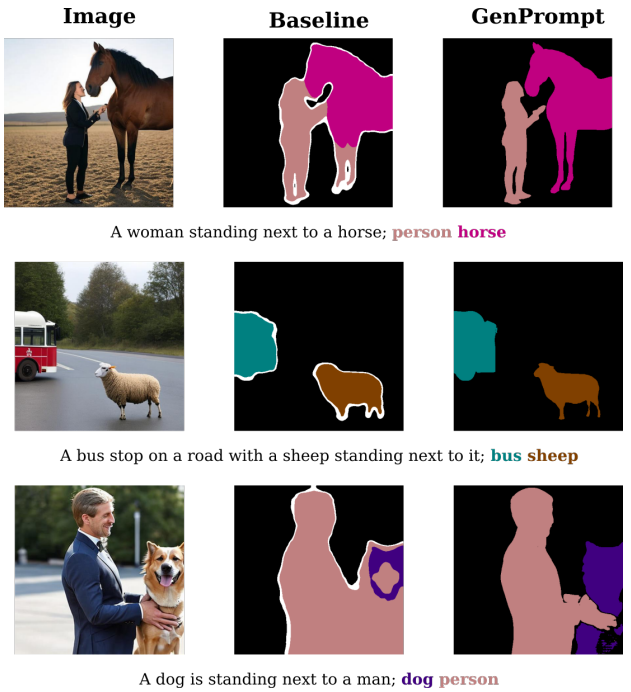


Figure 5. **Additional qualitative results, refer Sec. 3.3 for details.**

6.2. Additional qualitative results

We present additional qualitative results in Figures 5, 6 and 7. These examples showcase the superiority of our proposed prompting technique over the thresholding baseline.

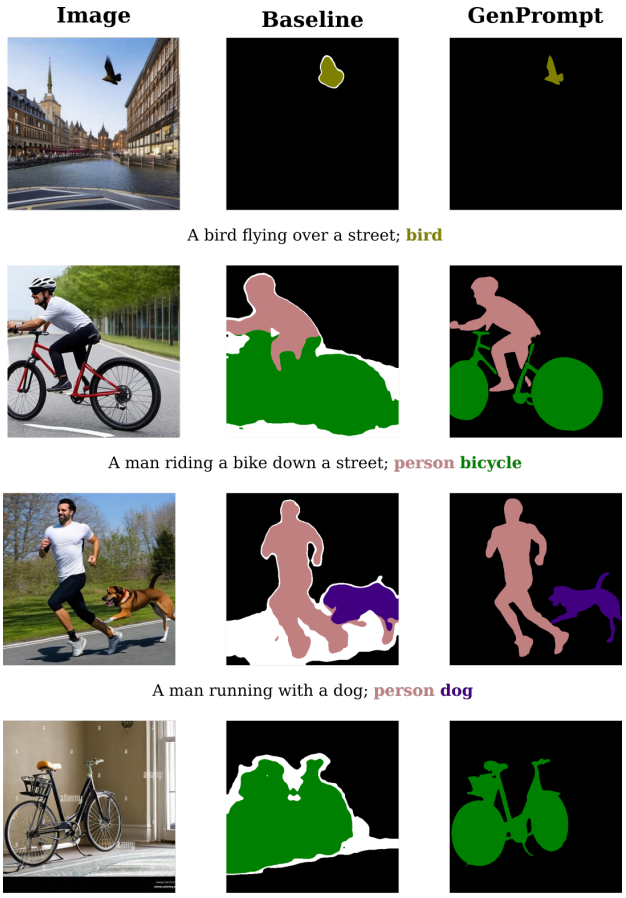


Figure 6. **Additional qualitative results, refer Sec. 3.3 for details.**

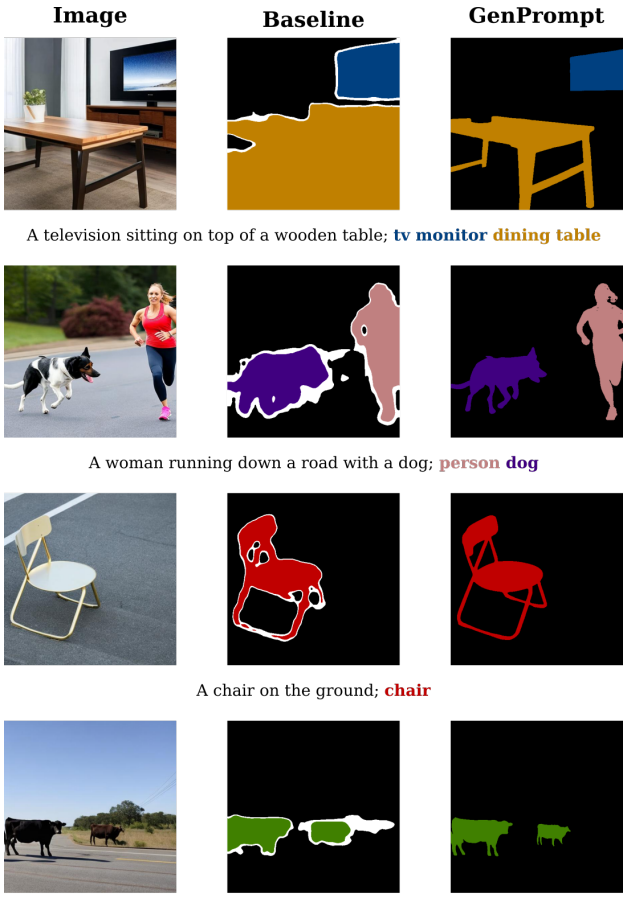


Figure 7. **Additional qualitative results, refer Sec. 3.3 for details.**