A. Supplementary

This supplementary section provides more information about our experiments, evaluation methods and additional quantitative and qualitative results. We describe in detail how we generate our two evaluation datasets and how we calculate the counting performance of our and previous approaches. We additionally provide more quantitative results to visualize the impact of the choice of our hyperparameters. Finally, we provide a further rich qualitative comparison of our method, Stable Diffusion and Attend-and-Excite to show that our approach outperforms existing ones in various scenarios.

A.1. Dataset

We create two separate datasets for measuring counting loss guidance evaluated by our counting metric and attention loss guidance evaluated by text-image/text-text similarity. The dataset for counting evaluation consists of prompts of a single object with a specific object count. We utilize the 34 object classes from Tab. 1, providing a good balance between simpler to generate objects like fruits and more complex objects like animals. We cover a broad range of object counts ranging from 1-20 per object class to test and compare our method to previous ones. We generate 680 prompts (20 different counts times 34 objects) with the template of the form "{count} {object}" to construct prompts like "one apple", "three lemons" and "six onions".

For evaluating our attention loss guidance we use the same 34 objects and build prompts containing two object classes per prompt. Specifically, we form object pairs by combining each object with each other disregarding order and create two prompts per pair with a random count for each object ranging from 1-20. This results in a total of 1122 prompts. We use the template "{count_a} {object_a} and {count_b} {object_b}" yielding examples like "ten cats and five birds", "nineteen birds and eight lemons" and "five elephants and twelve chicks".

Table 1. Dataset

Animals	cat, dog, bird, bear, lion, horse, elephant, monkey, frog, turtle, rabbit, mouse, chick
Objects	backpack, glasses, crown, suitcase, chair, balloon, bow, car, bowl, bench, clock, apple, banana, donut, orange, egg, tomato, lemon, macaron, bread, onion

A.2. Testing Environment

For our experiments, we use PyTorch [3] with a single NVIDIA Tesla V100 32GB GPU. It takes about 12 seconds to generate one image with vanilla Stable Diffusion, while our method takes about 26.9 seconds when using counting guidance for a single object. For two object classes it takes 15 seconds when using attention map guidance only and 37.6 seconds when using both attention map guidance and counting guidance.

A.3. Counting Metric

To calculate our counting metric, we use the state of the art pretrained object detection model Grounding DINO [1] with Swin-T [2] backbone to detect bounding boxes in the generated images. We use the fact that Grounding DINO is able to perform object detection with arbitrary class labels specified as prompts and thus use the objects in the prompt as detection classes. After detection, we count the number of output boxes per object class and compare it with the ground truth count in the prompt. To balance the influence of small and large object counts on the final metric, we additionally normalize our metric by the ground truth object count. Our normalized MAE metric for one object class is given as

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{\hat{y}_i - y_i}{y_i}\right|, \tag{1}$$

while our normalized RMSE metric is defined as

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{\hat{y}_i - y_i}{y_i}\right)^2}, \tag{2}$$

where y_i is the ground truth object count from the prompt and $\hat{y}_i$ is the number of detected bounding boxes in the generated image for the respective class.

A.4. Hyperparameter Analysis

Counting Loss Scale To determine the ideal counting loss scale, we run our method with various scales on our 680 prompts counting dataset and plot the resulting *MAE* and *RMSE* metrics in Figs. 1a and 1b. We choose $s_{count} = 1$ for our method (constant) since it provides a good value for both *MAE* and *RMSE*. As s_{count} increases, the counting error initially decreases but subsequently rises, exhibiting the behavior of a **convex function.** While excessive gradient guidance can negatively impact image generation, we demonstrate that increasing counting guidance up to a certain threshold can effectively reduce the counting error.

Fig. 1c shows the counting error (MAE) versus the number of objects N in the prompt for five s_{count} values, and Fig. 1d depicts its linear trend. As s_{count} increases, the slope of the linear trend gradually decreases. As a result, for small N, the performance is better when the s_{count} is smaller, while for large N, the performance improves as the s_{count} increases. This observed trend aligns with the intuition that increasing N poses greater challenges for accurate generation, thereby necessitating a larger s_{count}.

Our analysis yielded $s_{count} = \max(0.01, 0.2N - 1)$, which is a simple increasing function of N that significantly improves performance compared to a constant value.

Attention Loss Scale Similarly, we visualize the text-text and text-image similarity on our 1122 multi object class dataset for various attention loss scales in Fig. 2. We notice a strong peak of text-text similarity at the value 1 and thus choose our attention loss scale for our experiments as 1.

A.5. Additional Qualitative Results

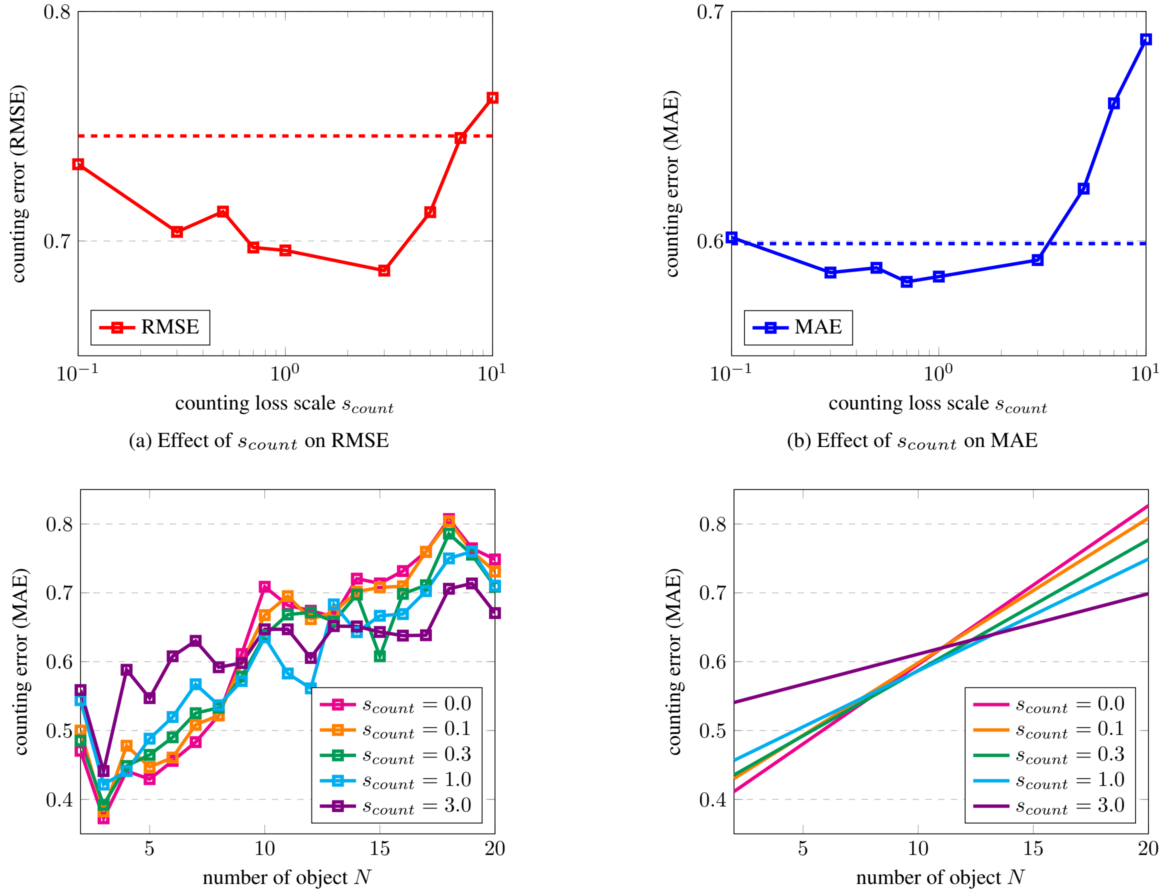
Fig. 3, Fig. 4 and Fig. 5 show additional results for our counting guidance with various prompts and varying object count for Stable Diffusion, Attend and Excite and ours. Even though we need to tweak our counting guidance scale hyperparameter for some prompts, our counting guidance method consistently creates the correct amount or, when dealing with large count, a similar amount of objects, whereas Stable Diffusion and Attend and Excite fail in many cases. When the object count grows, it becomes more challenging to generate the exact amount, however, our method nevertheless outperforms the other two tested methods.

Fig. 6 visualizes the attention map per object for several prompts for Stable Diffusion and our attention map guidance. We note that our attention maps capture the spatial location of each object more accurately than Stable Diffusion, while reducing the overlap between different objects.

A.6. Template for User Study and GPT Evaluation

User Study Compare the first and second images provided, and select the one that more closely aligns with the given prompt. Pay particular attention to the object count.

GPT Evaluation Prompt Compare the first and second images provided, and select the one that more closely aligns with the given prompt. Pay particular attention to the accuracy of the object count. Your selection can be subjective. Your final output score must be either 0 (if the first image is best), 0.5 ('Tie'), or 1 (if the second image is best). You have to give your output in this way (Keep your reasoning concise and short. Give your intermediate thinking step by step.)



(a) Effect of s_{count} on RMSE

(b) Effect of s_{count} on MAE

(c) Effect of N on MAE

(d) Effect of N (linear trend)

Figure 1. **Hyperparameter study.** Evaluated on 680 images.

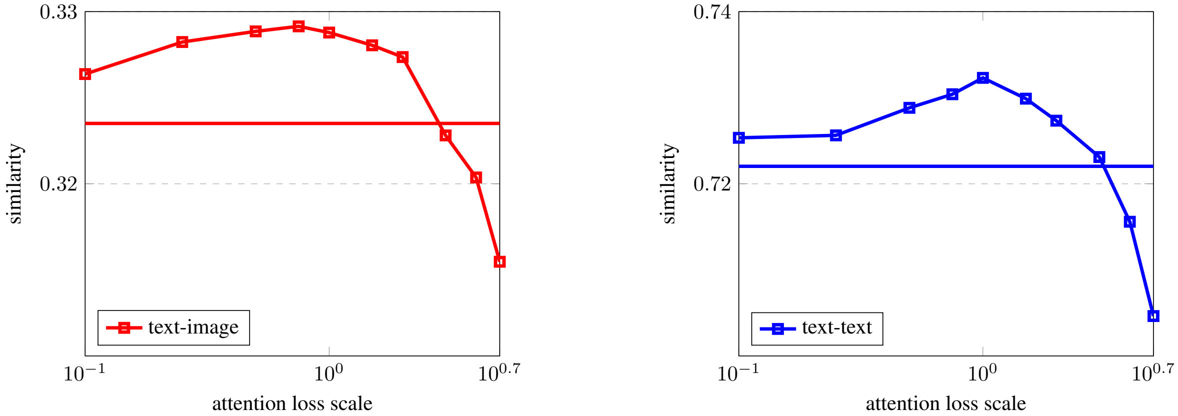
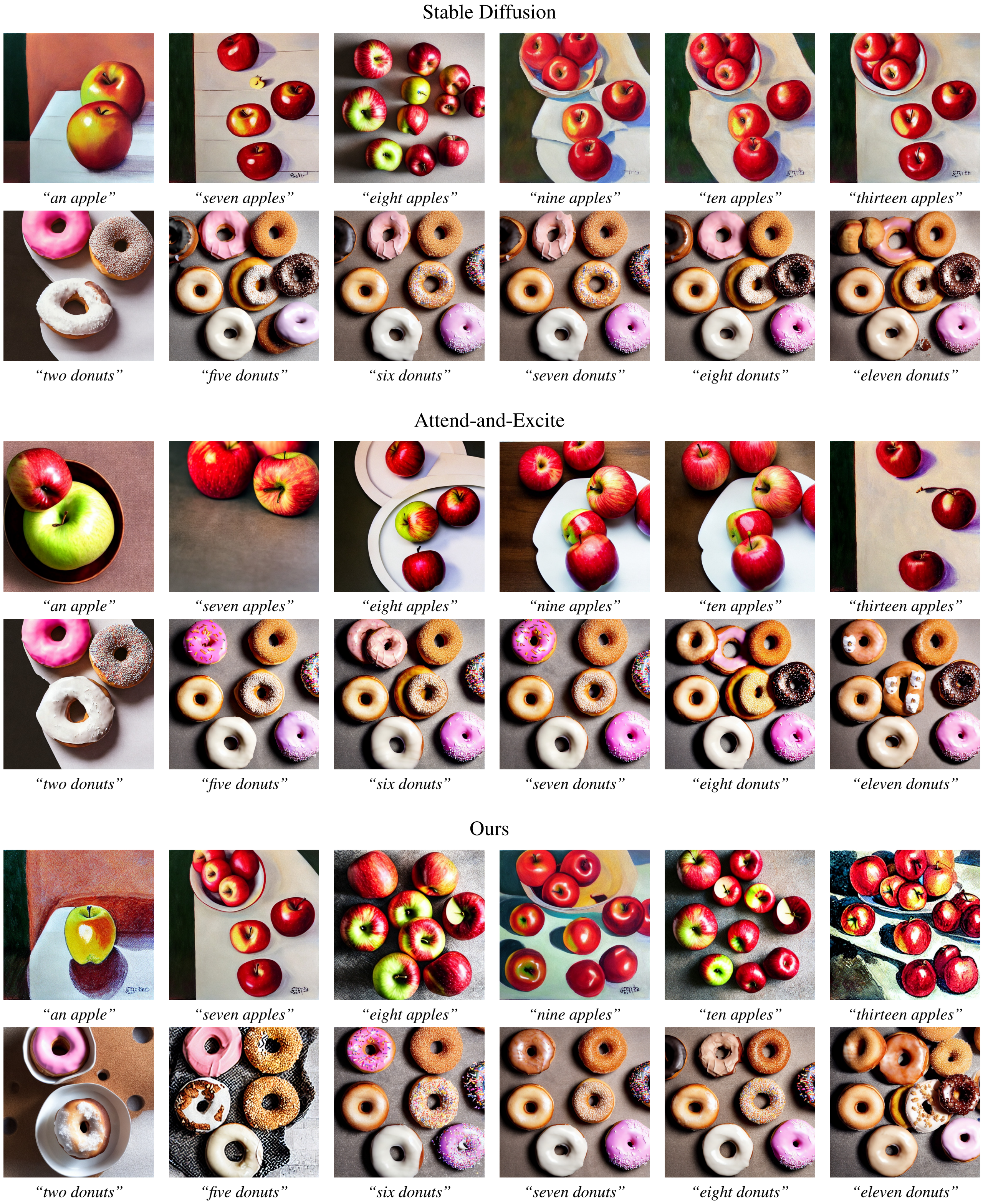


Figure 2. Effect of attention loss scale on the text-image and text-text CLIP similarity. Evaluated on our 1122 two object prompt dataset.



Stable Diffusion

“an apple” *“seven apples”* *“eight apples”* *“nine apples”* *“ten apples”* *“thirteen apples”*

“two donuts” *“five donuts”* *“six donuts”* *“seven donuts”* *“eight donuts”* *“eleven donuts”*

Attend-and-Excite

“an apple” *“seven apples”* *“eight apples”* *“nine apples”* *“ten apples”* *“thirteen apples”*

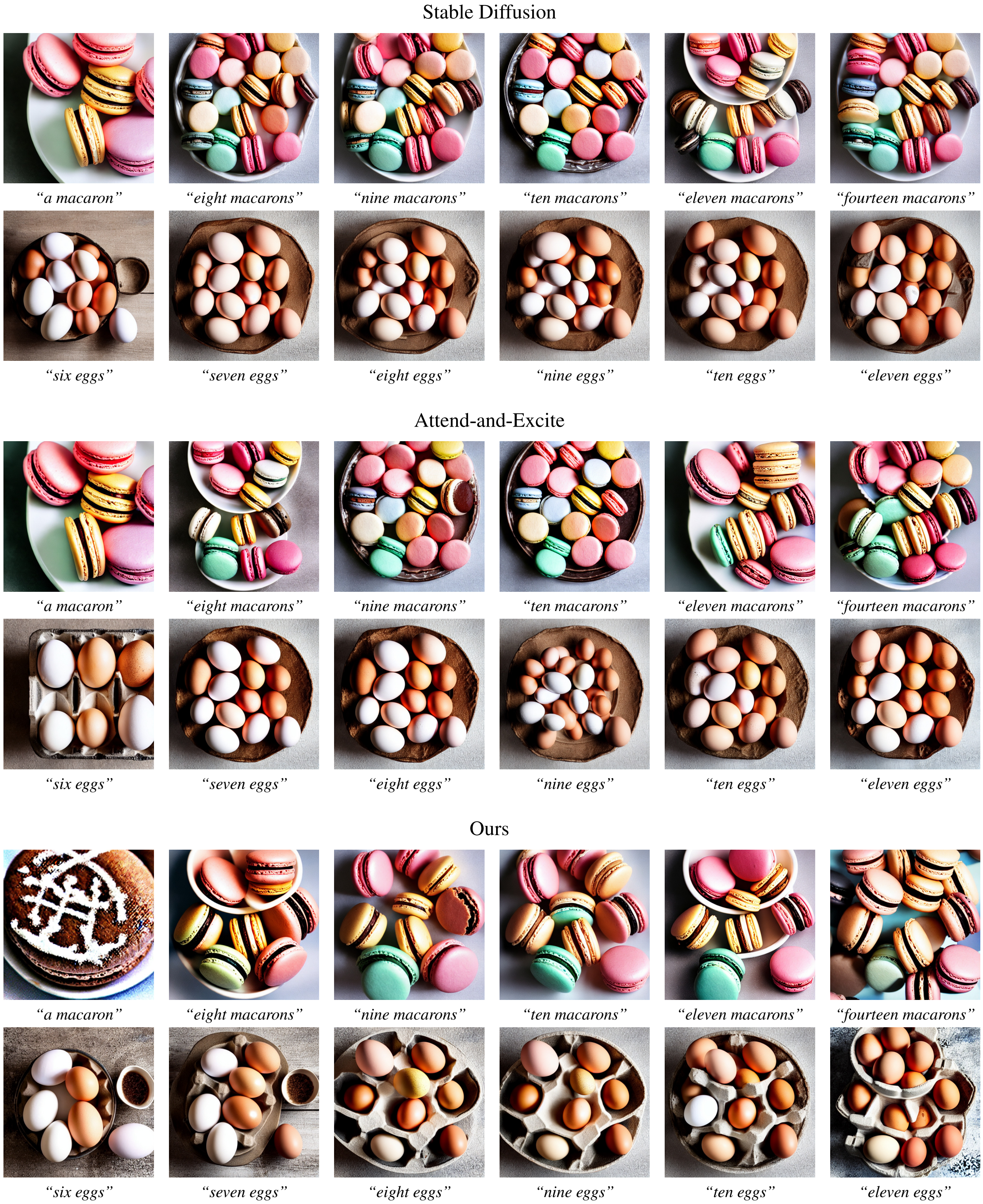
“two donuts” *“five donuts”* *“six donuts”* *“seven donuts”* *“eight donuts”* *“eleven donuts”*

Ours

“an apple” *“seven apples”* *“eight apples”* *“nine apples”* *“ten apples”* *“thirteen apples”*

“two donuts” *“five donuts”* *“six donuts”* *“seven donuts”* *“eight donuts”* *“eleven donuts”*

Figure 3. Additional qualitative results (1)



Stable Diffusion

"a macaron" *"eight macarons"* *"nine macarons"* *"ten macarons"* *"eleven macarons"* *"fourteen macarons"*

"six eggs" *"seven eggs"* *"eight eggs"* *"nine eggs"* *"ten eggs"* *"eleven eggs"*

Attend-and-Excite

"a macaron" *"eight macarons"* *"nine macarons"* *"ten macarons"* *"eleven macarons"* *"fourteen macarons"*

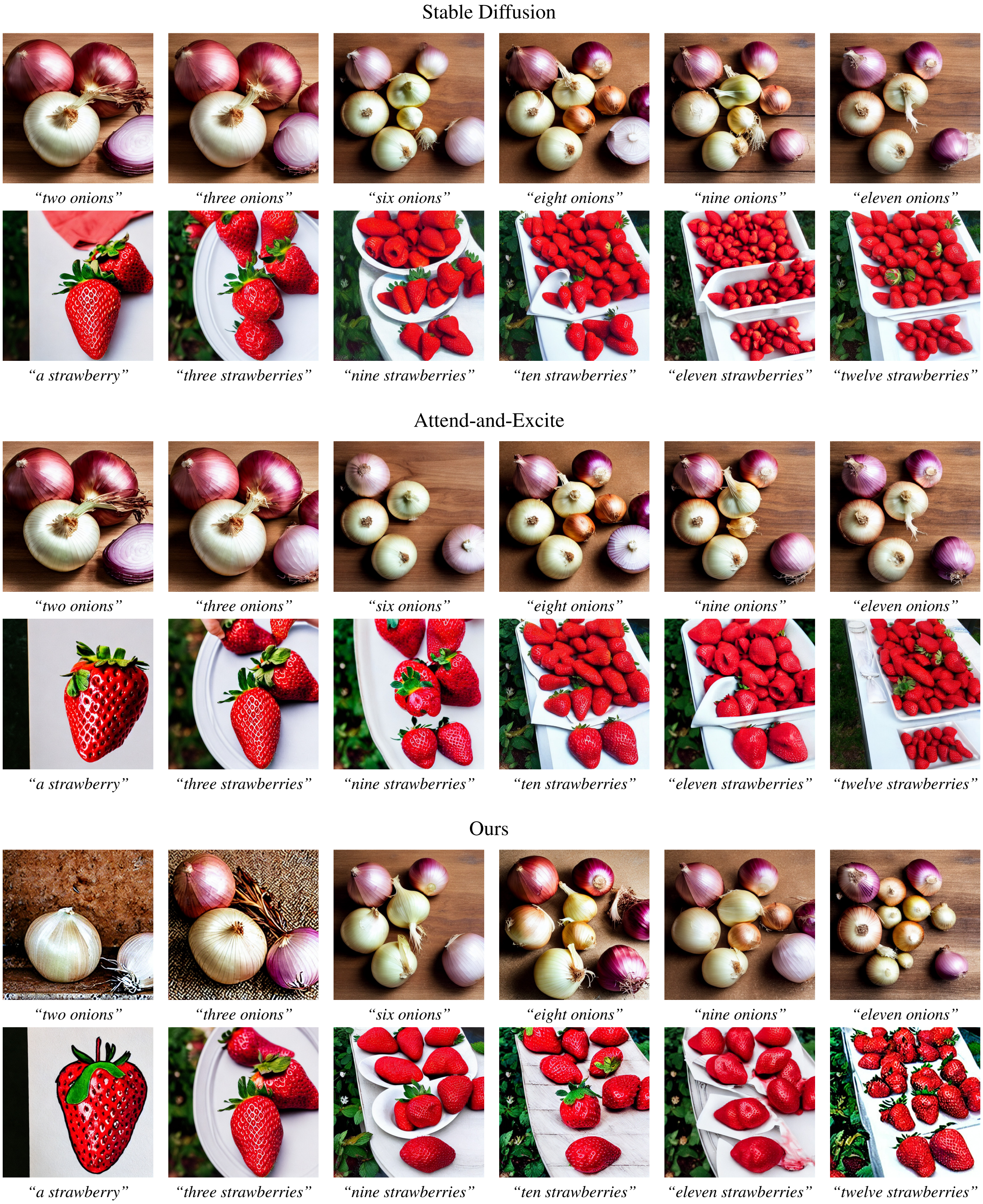
"six eggs" *"seven eggs"* *"eight eggs"* *"nine eggs"* *"ten eggs"* *"eleven eggs"*

Ours

"a macaron" *"eight macarons"* *"nine macarons"* *"ten macarons"* *"eleven macarons"* *"fourteen macarons"*

"six eggs" *"seven eggs"* *"eight eggs"* *"nine eggs"* *"ten eggs"* *"eleven eggs"*

Figure 4. Additional qualitative results (2)



Stable Diffusion

“two onions” *“three onions”* *“six onions”* *“eight onions”* *“nine onions”* *“eleven onions”*

“a strawberry” *“three strawberries”* *“nine strawberries”* *“ten strawberries”* *“eleven strawberries”* *“twelve strawberries”*

Attend-and-Excite

“two onions” *“three onions”* *“six onions”* *“eight onions”* *“nine onions”* *“eleven onions”*

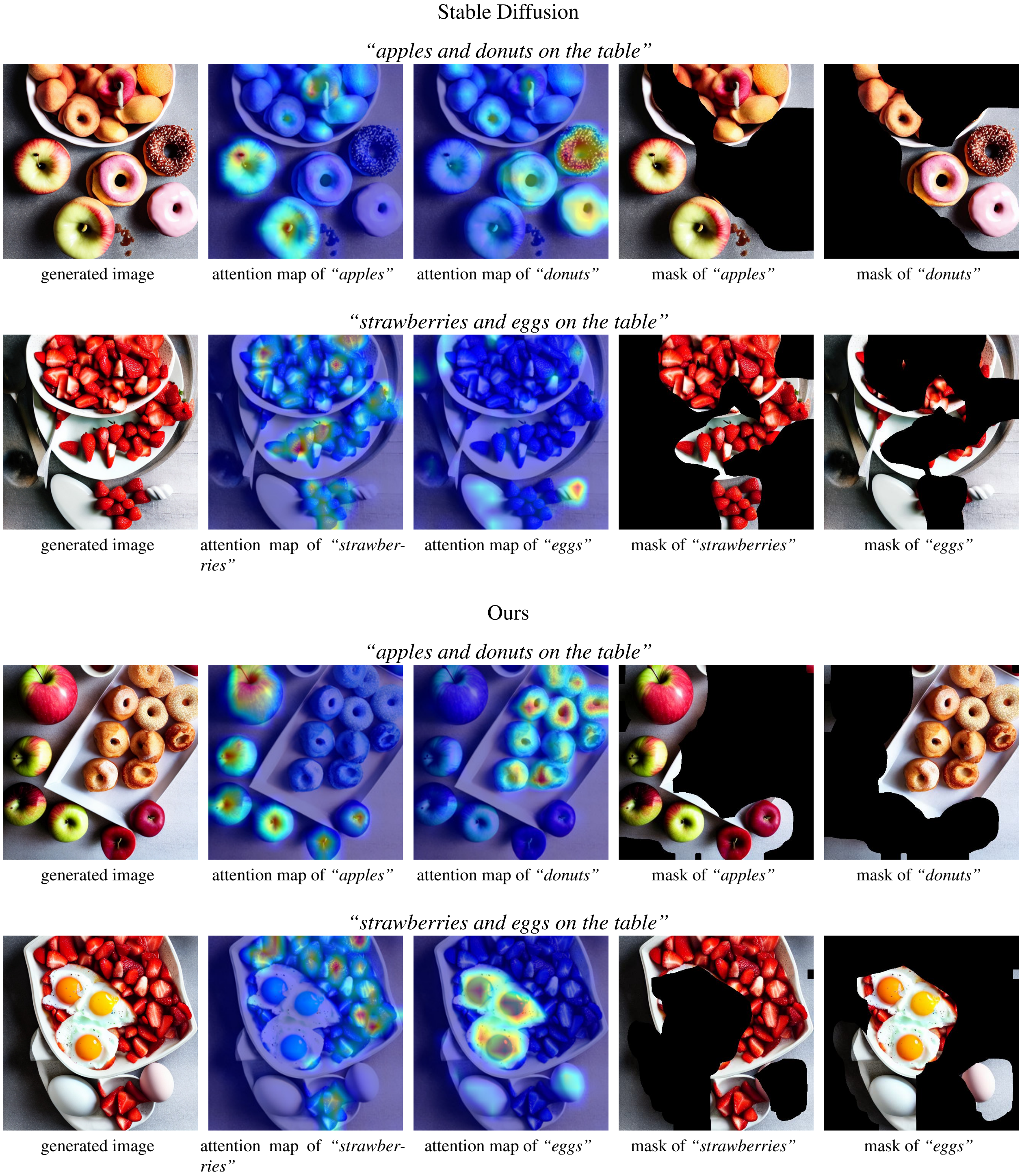
“a strawberry” *“three strawberries”* *“nine strawberries”* *“ten strawberries”* *“eleven strawberries”* *“twelve strawberries”*

Ours

“two onions” *“three onions”* *“six onions”* *“eight onions”* *“nine onions”* *“eleven onions”*

“a strawberry” *“three strawberries”* *“nine strawberries”* *“ten strawberries”* *“eleven strawberries”* *“twelve strawberries”*

Figure 5. Additional qualitative results (3)



Stable Diffusion

"apples and donuts on the table"

generated image | attention map of *"apples"* | attention map of *"donuts"* | mask of *"apples"* | mask of *"donuts"*

"strawberries and eggs on the table"

generated image | attention map of *"strawberries"* | attention map of *"eggs"* | mask of *"strawberries"* | mask of *"eggs"*

Ours

"apples and donuts on the table"

generated image | attention map of *"apples"* | attention map of *"donuts"* | mask of *"apples"* | mask of *"donuts"*

"strawberries and eggs on the table"

generated image | attention map of *"strawberries"* | attention map of *"eggs"* | mask of *"strawberries"* | mask of *"eggs"*

Figure 6. Additional qualitative results (4)

References

[1] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, and Lei Zhang. Grounding dino: Marrying dino with grounded pre-training for open-set object detection, 2023. 1

[2] 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 *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 1

[3] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. 1