Attention, May I Have Your Decision? Localizing Generative Choices in Diffusion Models

Supplementary Material

A. Implementation details on Steering vectors

To calculate the steering vectors discussed in Section 3.2, we use a pipeline composed of a StandardScaler[1] and a LogisticRegression[2] model (with a maximum of 1000 iterations) on the pooled activations for each layer–timestep pair. After training, we map the learned coefficients back to the original, unstandardized feature space by dividing the weight vector by the corresponding scaling factors. The rescaled vector is then normalized to unit length:

$$s_{\ell,t} = \frac{\hat{w}_{\ell,t}}{\|\hat{w}_{\ell,t}\|_2}.$$

yielding the final steering vector $s_{\ell,t}$ for layer ℓ and timestep t. The vector is used later during generation as:

$$H'_{\ell,t} = H_{\ell,t} + \alpha\, s_{\ell,t},$$

where $H_{\ell,t}$ denotes the unmodified activation tensor and the scaling factor α controls the magnitude of the intervention. We design the probes in a binary fashion. For example, if the vector is trained with *old* as the positive class, positive α values shift the generation toward older appearances, whereas negative α values shift it toward not-old ones. The magnitude of α determines the strength of the modification, as illustrated in Figure 8. In the backward diffusion pass with classifier-free guidance, the steering vector is applied only to the component conditioned on the text prompt.

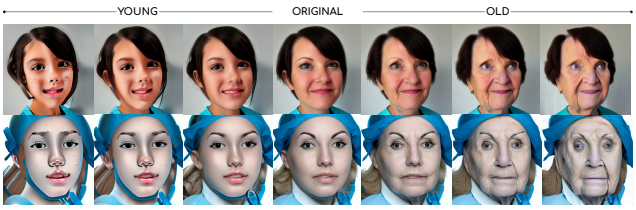

Figure 8. Effect of increasing α values along the young–old direction. Larger α produces stronger age-related changes.

B. Additional details on experimental Setup

For the steering-based debiasing, we introduce a random component that selects the direction in which the entire batch of images is shifted. When there are n possible decisions, each direction is chosen with probability $\frac{1}{n}$. For example, for race we consider four options - white, black, asian, and indian - so each has a probability of 0.25.

We use steering vectors trained as binary classifiers for the following pairs: woman–man, young–old, white–black, white–indian, and white–asian. Since most generated images are classified as adult or white, we do not apply modifications for the adult or white directions. For binary directions (e.g., age), the steering vector supports both positive and negative α values, enabling movement toward either side of the decision boundary. All selected layers and their corresponding α values are summarized in Table 6.

Table 6. Parameters used for steering across different directions.

Direction	Layers	α
woman	up_blocks.1.attn.2.t_blocks.0.attn1 up_blocks.1.attn.1.t_blocks.0.attn1 up_blocks.1.attn.0.t_blocks.0.attn1 mid_block.attn.0.t_blocks.0.attn1	-10
man	up_blocks.1.attn.2.t_blocks.0.attn1 up_blocks.1.attn.1.t_blocks.0.attn1 up_blocks.1.attn.0.t_blocks.0.attn1 mid_block.attn.0.t_blocks.0.attn1	10
old	up_blocks.1.attn.1.t_blocks.0.attn1 mid_block.attn.0.t_blocks.0.attn1 up_blocks.1.attn.0.t_blocks.0.attn1	8
young	up_blocks.1.attn.1.t_blocks.0.attn1 mid_block.attn.0.t_blocks.0.attn1 up_blocks.1.attn.0.t_blocks.0.attn1	-8
adult	–	–
black	up_blocks.1.attn.1.t_blocks.0.attn1 mid_block.attn.0.t_blocks.0.attn1	15
indian	mid_block.attn.0.t_blocks.0.attn1 up_blocks.1.attn.2.t_blocks.0.attn1 down_blocks.2.attn.0.t_blocks.0.attn1	10
asian	up_blocks.1.attn.1.t_blocks.0.attn1 mid_block.attn.0.t_blocks.0.attn1 up_blocks.1.attn.0.t_blocks.0.attn1	10
white	–	–

We apply each modification only after the first 15 timesteps, as the logistic-regression classifiers exhibit lower accuracy during the initial stages of denoising. Figure 10 shows the test accuracy across several selected layers, illustrating that accuracy increases after the early timesteps. Applying the intervention later in the denoising process better preserves the structure of the generated images, as shown in Figure 9.

C. Additional details on fine-tuning

We finetune Stable Diffusion v1.5 using low-rank adaptation (LoRA), applying rank-32 (for gender concept in main results) and rank-64 (for age and race) adapters to selected layers while

[1] scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html

[2] scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

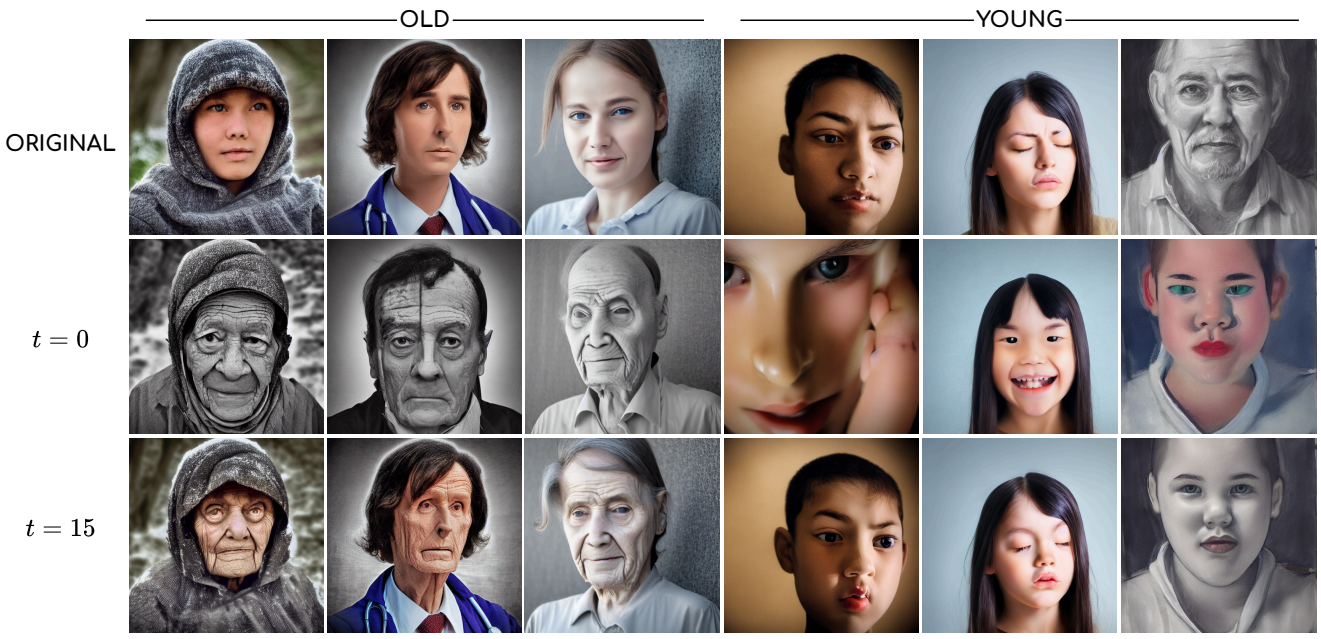

Figure 9. Example generations showing that applying the steering vector at later timesteps preserves the overall image structure while still shifting the predicted age direction. Here, t denotes the timestep at which the modification begins.

Table 7. Finetuning comparison for cross-attention layers selected via prompt injection versus randomly chosen layers.

Method	**Gender (2)**				**Age (3)**				**Race (4)**			
	FD ↓	FID ↓	CLIP-I ↑	CLIP-T ↑	FD ↓	FID ↓	CLIP-I ↑	CLIP-T ↑	FD ↓	FID ↓	CLIP-I ↑	CLIP-T ↑
Original	0.564	120.06	-	0.6155	0.752	120.06	-	0.6155	0.558	120.06	-	0.6155
Finetuning (rank=32, selected)	**0.515**	**128.27**	0.9028	0.6172	**0.699**	**114.34**	**0.8943**	0.6173	**0.485**	127.90	0.9062	**0.6190**
Finetuning (rank=32, random)	0.542	133.62	**0.9230**	**0.6177**	0.733	117.61	0.8916	**0.6191**	0.537	**123.64**	**0.9412**	0.6159

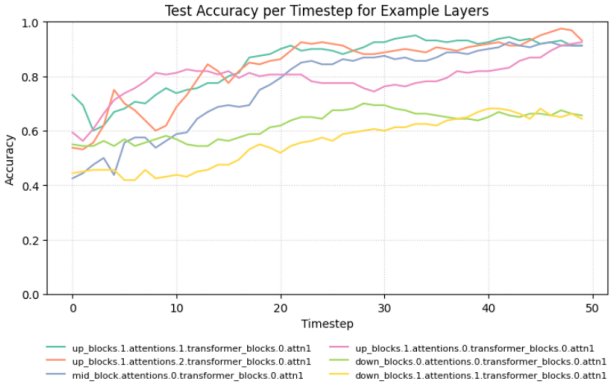

Figure 10. Test accuracy across the selected layers and timesteps for gender dataset.

keeping all other parameters frozen. We optimize the model with AdamW using a learning rate of 3×10^{-5}, a cosine learning rate schedule, and 1000 warmup steps. Training is performed with mixed-precision `bf16`, gradient clipping with a maximum norm of 1, and gradient checkpointing. We resize images to a resolution of 512×512 with center cropping and random horizontal flips. We use a batch size of 2 with 4 gradient accumulation steps. The model is trained for 30 epochs with 8 data-loader workers. To align with previous works, in ICM(Finetuning), we finetune the model using only images generated by the model itself, which directly impacts the performance of the finetuned model in terms of FID.

For the main debiasing comparison, gender and age fine-tuning used the same layers as for steering; race used the union of all layers from the black, Indian, and Asian experiments (Table 6).

We compare fine-tuning only a selected subset of layers identified by prompt injection (described in section H) against fine-tuning random layers. The results in Table 7 show that updating only the selected layers achieves stronger performance on most metrics.

D. Evaluation metrics

We measure bias using Fairness Discrepancy (FD) [26, 35], which is the Euclidean distance between a reference and a generated distribution:

$$\mathrm{FD} = \left\| \bar{p} - \mathbb{E}_{\mathbf{x} \sim p_\theta(\mathbf{x})}(\mathbf{y}) \right\|_2 \tag{3}$$

where $\bar{p}$ is the reference distribution, $\mathbf{x}$ are model samples, and

$\mathbf{y}$ is their predicted attribute distribution. A lower FD value means the generated distribution is closer to the reference. For Stable Diffusion v1.5, we generate 500 images per prompt, compute metrics for each, and report the average. FairFace [17] provides predictions for gender, age, and race. Following prior work, we use two gender classes (male, female), three age groups (young: 0–19, adult: 20–59, old: 60+), and four race groups (white, black, Asian - combining East and Southeast Asian - and Indian).

We assess image quality using the Fréchet Inception Distance (FID), with the FFHQ [13] dataset as the reference set of real images. FID measures the distance between generated and real image distributions, with lower values indicating closer alignment.

We also use CLIP-I to assess alignment with reference images and CLIP-T to assess alignment with the input prompts. For this, we generated original image embeddings $\mathbf{e}_{\text{img}}^{\text{orig}}$ and debiased image embedding $\mathbf{e}_{\text{img}}^{\text{gen}}$ to calculate CLIP-I metric:

$$\text{CLIP-I} = \frac{\mathbf{e}_{\text{img}}^{\text{orig}} \cdot \mathbf{e}_{\text{img}}^{\text{gen}}}{\|\mathbf{e}_{\text{img}}^{\text{orig}}\| \, \|\mathbf{e}_{\text{img}}^{\text{gen}}\|}. \tag{4}$$

For textual alignment, we use text prompt embedding $\mathbf{e}_{\text{text}}$ and debiased images $\mathbf{e}_{\text{img}}^{\text{gen}}$:

$$\text{CLIP-T} = \frac{\mathbf{e}_{\text{text}} \cdot \mathbf{e}_{\text{img}}^{\text{gen}}}{\|\mathbf{e}_{\text{text}}\| \, \|\mathbf{e}_{\text{img}}^{\text{gen}}\|}. \tag{5}$$

For embedding extraction, we use the CLIP ViT-L/14 model [30].

E. Details on prompt templates

For training logistic regression models, we have two prompt versions: (1) a general prompt without decision information, and (2) a specific prompt containing direct information.

Examples of general prompts:

a face of a person
photo of a person
portrait photo of a person
a close-up face of a person
a detailed portrait of a person
studio photo of a person
professional photo of a person
a headshot of a person
a professional studio headshot of a person
portrait of a person in natural light
portrait of a person indoors

Examples of specific prompts for gender:

a portrait of a **woman** in natural light
photo of a **man** wearing casual clothes
a detailed portrait of a **lady** with soft lighting
a close-up face of a **gentleman** with calm expression
portrait of a **girl** outdoors in daylight
a headshot of a **woman** against a plain background
studio photo of a **man** with neutral expression
portrait of a **boy** with gentle expression
photo of a **woman** taken in warm sunset light
a professional photo of a **man** in front of a window
portrait of a **lady** with confident look

F. Additional qualitative results

We observe that training logistic regression classifiers on general prompts yields less artificial, more natural images than training on highly specific prompts. Figure 11 shows outputs from the original model and from steering vectors trained on general or specific prompts, all using the same α.

Figure 11. Comparison of generations from the original model (top), steering trained on general prompts (middle), and steering trained on specific prompts (bottom).

We compared several α configurations and observed that stronger values tend to improve the debiasing performance. However, as illustrated in Figure 12, excessively large α values introduce visible artifacts and may degrade the overall visual quality of the generated images.

In Figure 13 we present the qualitative ablation using FLUX. We evaluate both double and single stream blocks to identify the most effective layers, applying steering to the image part of each block's output. After steering 10 selected layers, the target class distribution increased from the initial 61% to 98% for "woman" and from 39% to 95% for "man", while maintaining CLIP-I scores of 0.869 and 0.831, respectively, demonstrating the generalization of our method to MM-DiT-based architectures.

To evaluate our method in more complex scenarios, we experiment with pose modification. Figure 14 shows the effect of steering with the 10 selected layers on the dog's pose.

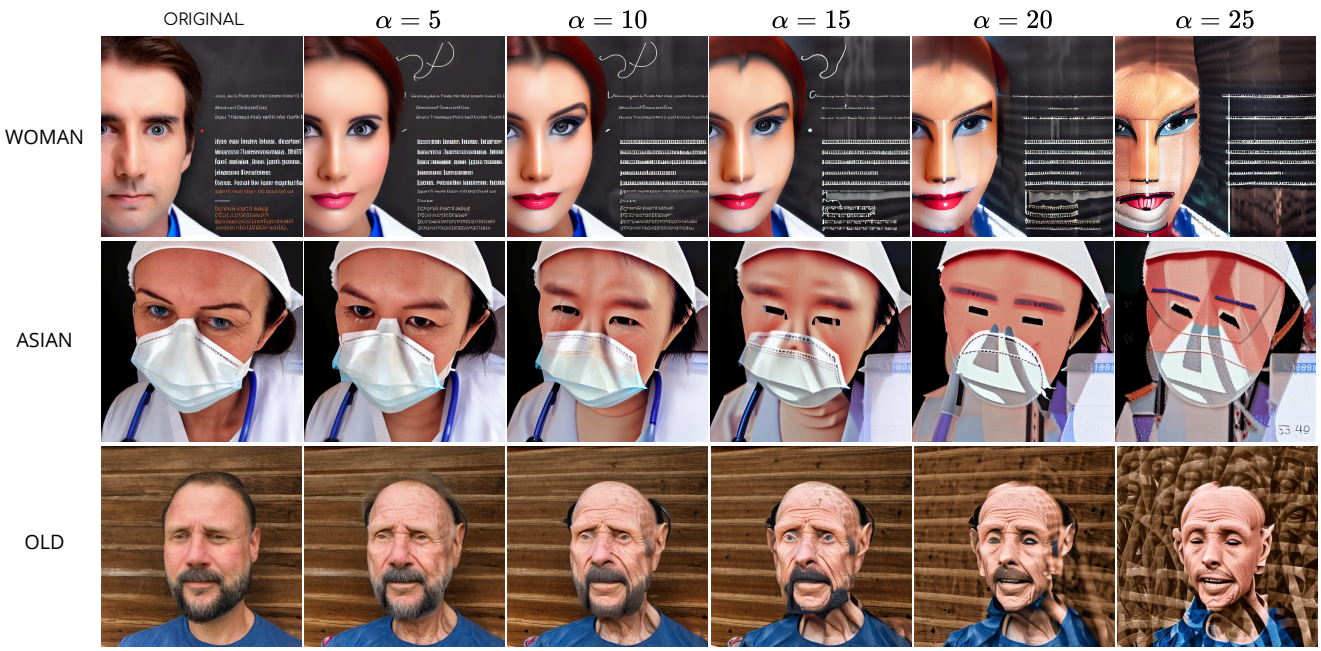

Figure 12. Example generations showing how increasing the steering strength α can progressively degrade the output.

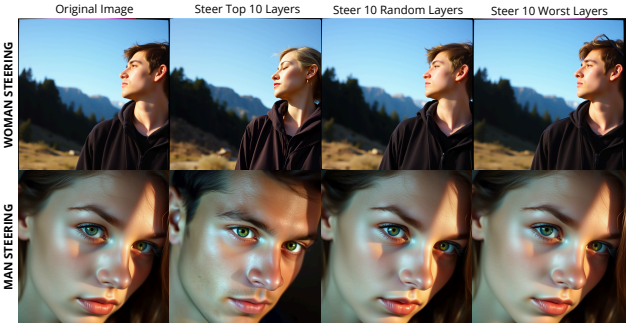

Figure 13. FLUX steering.

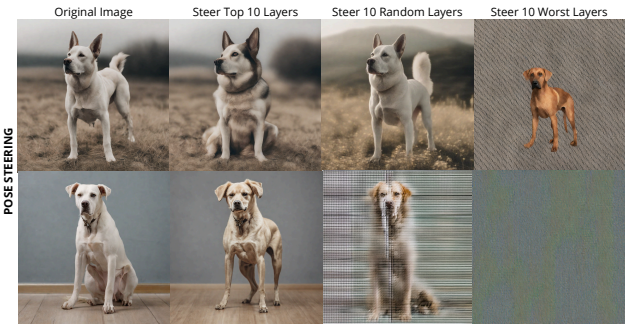

Figure 14. SDXL steering.

G. Scalability

While our approach involves extensive linear probing, the process is computationally efficient (∼14 minutes on a 288-core CPU). We can achieve a $10\times$ speedup by utilizing a single steering vector derived from five steps; this optimized workflow yields nearly identical performance, with an FD of 0.08 and a $CLIP\text{-}I$ score of 0.89 for gender debiasing.

We use average pooling primarily for computational feasibility, as using raw activations drastically increases the number of examples and memory requirements (e.g., for activations of shape $(1024, 640)$, $1024\times$ more vectors). Since we calculate probes over 3k samples, using raw activations would result in the infeasible dataset of $\approx$ 183 GB (for a single layer only), compared to $\approx$ 0.2 GB with average pooling. However, we run an additional experiment by sampling 10 random patches from each activation, achieving slightly worse results compared to the average-pooled representation: FD=0.136 (vs 0.094) and CLIP-I=0.864 (vs 0.879).

H. Prompt injection experiment details

In our experiments, we compare our localization approach with a prompt-injection-based approach, focusing on localizing social attributes related to age, gender, and race across cross-attention layers. For each target decision D, we construct a decision-specific dataset consisting of general prompts $\{p_{gen,1}, p_{gen,2}, \ldots, p_{gen,N}\}$ (e.g., *portrait of a doctor*). Each general prompt is paired with a collection of specific prompts $\{p_{spec,i,1}, p_{spec,i,2}, \ldots, p_{spec,i,M}\}$ that enumerate the possible outcomes for the target attribute (e.g., for gender: *image of a woman*).

We provide the templates used to construct general prompts by inserting profession names:

portrait of a {}
face of a {}
a realistic portrait photo of a {} looking at the camera
a well-lit studio portrait of a {} with sharp focus
{} captured in a professional headshot
a {} at work, close-up portrait
a professional close-up headshot of a {} in uniform
a {} concentrating on their work, upper-body view
a face photograph of a {} with a neutral expression
a full-body photograph of a {} at work

For each provided prompt template, we use job names associated with women: *nurse*, *housekeeper*, *receptionist*, *secretary*, and *librarian*. Male-associated professions are *construction worker*, *doctor*, *lawyer*, *farmer*, and *CEO*.

During each image generation, we inject a specific prompt into a single cross-attention layer across all timesteps, while all remaining layers receive the general prompt. For each prompt, we generate three images using three different seeds. We then compare the outputs with the attribute indicated by the specific prompt (e.g., *image of a woman* or *image of a man* when analyzing gender) to assess how the model's decision changes. In this way, we identify the top k layers that have the strongest impact on the final results. An example of prompt injection, where modifying a single layer changes the output from man to woman, is shown in Figure 15.

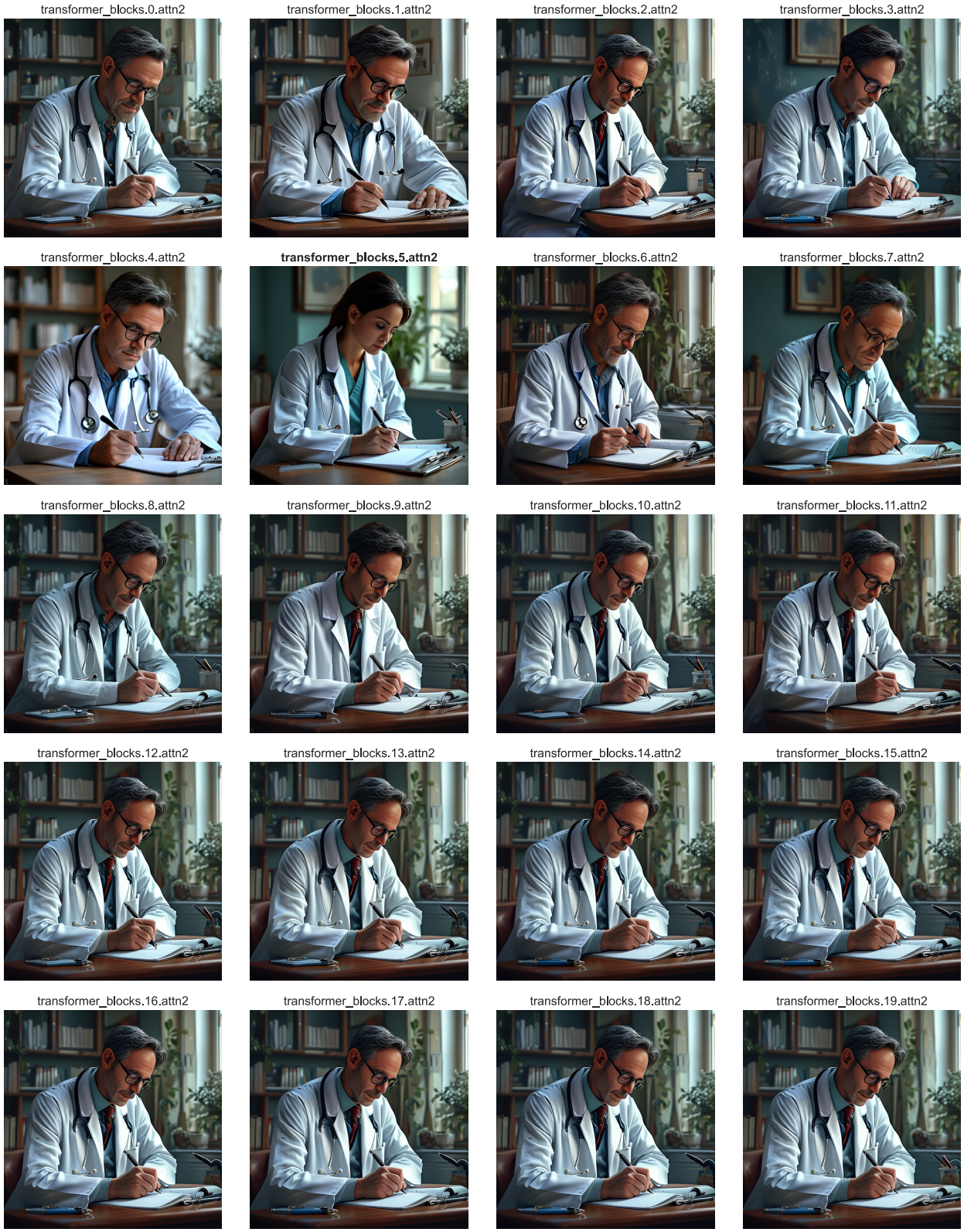

Figure 15. Example SANA images generated after injecting a specific prompt into a chosen cross-attention layer. The general prompt is *"A realistic photo of a doctor sitting and writing notes"*, and the specific prompt is *"a photo of a woman"*.