

A Definitions and Background

A.1 Preliminaries

Background on diffusion. Consider a data point \mathbf{x} drawn from the distribution $p_{data}(\mathbf{x})$, and let t be a time step within the interval $[0, 1]$. The forward process is defined as $\mathbf{x}_t = \mathbf{x} + \sigma(t)\epsilon$, where Gaussian noise $\epsilon \sim N(0, I)$ is added to the data. The function $\sigma(t)$ is monotonically increasing, with boundary conditions $\sigma(0) = 0$ and $\sigma(1) = \sigma_{max}$, where σ_{max} is significantly larger than σ_{data} . [Karras et al. \(2022\)](#) demonstrated that the progression of the noisy samples \mathbf{x}_t can be captured by an ordinary differential equation (ODE):

$$d\mathbf{x} = -\dot{\sigma}(t)\sigma(t)\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t)dt. \quad (3)$$

Alternatively, this can be expressed as a stochastic differential equation (SDE):

$$\begin{aligned} d\mathbf{x} = & -\dot{\sigma}(t)\sigma(t)\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t)dt \\ & - \beta(t)\sigma(t)^2\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t) + \sqrt{2\beta(t)\sigma(t)}d\boldsymbol{\omega}_t. \end{aligned} \quad (4)$$

In this context, $d\boldsymbol{\omega}_t$ represents the standard Wiener process, and $p_t(\mathbf{x}_t)$ denotes the distribution of the perturbed samples, with initial and final conditions $p_0 = p_{data}$ and $p_1 = \mathcal{N}(0, \sigma_{max}^2 I)$, respectively.

To sample from diffusion models, one must solve the diffusion ODE or SDE in reverse, moving from $t = 1$ back to $t = 0$. This process relies on the time-dependent score function $\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t)$, which is approximated by a denoiser $v_\theta(\mathbf{x}_t, t)$. This denoiser is trained to predict the original clean samples \mathbf{x} from their noisy counterparts \mathbf{x}_t . The framework also supports conditional generation by employing a denoiser $v_\theta(\mathbf{x}, t, y)$ that incorporates additional input signals y , such as class labels or textual prompts, allowing for more controlled and specific sample generation.

Background on flow matching. Flow matching approaches aim to learn a velocity field v_t mapping random noise $\mathbf{x}_0 = \epsilon \sim \mathcal{N}(0, I)$ to data samples $\mathbf{x}_1 \sim p_{data}(\mathbf{x})$. Such a mapping is obtained by solving an ordinary differential equation (ODE) of the form:

$$\frac{d\mathbf{x}_t}{dt} = v_t(\mathbf{x}_t). \quad (5)$$

[Lipman et al. \(2023\)](#) provide a simple simulation-free training objective for flow generative models by directly regressing the velocity field v_t on a conditional vector field $u_t(\cdot|\mathbf{x}_1)$:

$$\mathbb{E}_{t, q(\mathbf{x}_1), p_t(\mathbf{x}_t|\mathbf{x}_1)} \|v_t(\mathbf{x}_t) - \mathbf{u}_t(\mathbf{x}_t|\mathbf{x}_1)\|^2, \quad (6)$$

where $\mathbf{u}_t(\cdot|\mathbf{x}_1)$ uniquely determines a conditional probability path $p_t(\cdot|\mathbf{x}_1)$ towards data sample \mathbf{x}_1 . A popular choice for the conditional probability path corresponds to a linear interpolation between data and noise $\forall t \in [0, 1]$, $\mathbf{x}_t = t\mathbf{x}_1 + (1-t)\mathbf{x}_0$ resulting in a conditional probability path of the form $\mathbf{u}(\mathbf{x}_t|\mathbf{x}_1) = \mathbf{x}_1 - \mathbf{x}_0$. Once the conditional probability path is learned, sampling from the model can be achieved by solving the ODE defined in Equation (5) using any appropriate ODE solver from the literature.

Latent diffusion models (LDMs). [Rombach et al. \(2022\)](#) proposed to train diffusion models in a latent space induced by a pretrained and frozen autoencoder. The autoencoder converts all images into latents of smaller resolution, inducing a space in which the generative model is trained, afterwards the latents can be converted back into image space by using the decoder of the autoencoder. Such models allow to significantly scale up model training as the effective number of tokens representing each data point is reduced significantly. For example, recent autoencoders have a downscaling factor of 8×8 , resulting in a reduction of the sequence length by a factor of 64.

A.2 Rank scoring

As different guidance methods require their own set of hyperparameters, and may respond differently to shared hyperparameters, we select the best hyperparameter set for each method using a global score. The global score is computed as the average ranking across all tracked metrics, where the ranking is determined within the pool of tested hyperparameter combinations.

B Limitations

While our method is successfully able to boost the Pareto fronts for quality-consistency-diversity, tradeoffs between these metrics still subsist, although to a lesser extent, when using ERG. Future work should explore whether these tradeoffs are inherent to the model or a result of subpar sampling methods.

Additionally, our method operates under the assumption that the model follows the diffusion transformer architecture. As our method manipulates the energy landscape in attention layers, it needs to be tuned for different architectures containing a different number of attention layers. It is therefore not directly applicable to models that do not make use of attention layers in their architecture, such as U-Nets without self-attention.

Our method was only tested on diffusion transformers with standard architecture shapes (presenting a depth between 16 and 38), therefore extrapolating our claims to extremely deep/shallow models is not evident without supporting experiments. However, we believe such cases to be less relevant currently as they are not usual choices in state-of-the-art generative models.

C Interpreting ERG

In the following, we provide two interpretations of ERG based on entropy regularized RL and variational inference. Let p be the density function for the strong model and p^τ the one induced by the attention rectification mechanism we introduce in the paper.

I-ERG. We begin with the I-ERG guidance update, which modifies the model’s score estimate as:

$$\nabla_z \log p^\tau(\mathbf{z}) = \nabla_z \log p(\mathbf{z}) + w (\nabla_z \log \rho(\mathbf{z}) - \nabla_z \log p^\tau(\mathbf{z})). \quad (7)$$

Rewriting the right-hand side, we obtain:

$$\nabla_z \log p^\tau(\mathbf{z}) = \nabla_z \log p(\mathbf{z}) + w \nabla_z \log \left(\frac{\rho(\mathbf{z})}{p^\tau(\mathbf{z})} \right). \quad (8)$$

Integrating both sides with respect to \mathbf{z} , we find $p^\tau(\mathbf{x}) \propto p(\mathbf{x}) \cdot \exp(w \cdot R(\mathbf{x}))$, where $R(\mathbf{x}) = \log \left(\frac{\rho(\mathbf{x})}{p(\mathbf{x})} \right)$.

This form reveals that I-ERG guidance parallels a posterior distribution that re-weights the base model’s density $p(\mathbf{x})$ by an exponential function of the reward signal $R(\mathbf{x})$. This exponential reweighting aligns with objectives used in *entropy-regularized RL* or *KL control*, where the optimal distribution maximizes an expected reward under a KL penalty with respect to a prior:

$$\pi^* = \arg \max_{\pi} \mathbb{E}_{x \sim \pi} [R(\mathbf{x})] - \frac{1}{\lambda} \text{KL}(\pi || p), \quad (9)$$

where λ serves as an inverse temperature controlling the sharpness of reward influence. The solution to this optimization problem has the form:

$$\pi(\mathbf{x}) \propto p(\mathbf{x}) \cdot \exp(\lambda R(\mathbf{x})). \quad (10)$$

This shows that the I-ERG update implicitly implements the maximum entropy policy framework, steering the sampling distribution toward high-reward regions while maintaining proximity to the base model $p(\mathbf{z})$. Equivalently, using Bayes rule, it can be viewed as defining a *joint distribution* over \mathbf{x} under $p(\mathbf{x})$ and the energy induced by reward $R(\mathbf{x})$ (assuming conditional independence).

C-ERG. In the conditional setting, assume we start from a model $p(\mathbf{z}|\mathbf{s})$ conditioned on some semantic signal \mathbf{s} . Let $\mathbf{s}^\tau = \mathbf{s} + \Delta \mathbf{s}$ represent a degraded or weakened version of the conditioning (e.g., via temperature scaling or perturbation). The C-ERG guidance update is given by:

$$\nabla_z \log p^{\text{ERG}}(\mathbf{z} | \mathbf{s}, \mathbf{s}^\tau) = \nabla_z \log p(\mathbf{z} | \mathbf{s}) + w \cdot (\nabla_z \log p(\mathbf{z} | \mathbf{s}) - \nabla_z \log p(\mathbf{z} | \mathbf{s}^\tau)). \quad (11)$$

Or, equivalently:

$$\nabla_z \log p^{\text{ERG}}(\mathbf{z} | \mathbf{s}, \mathbf{s}^\tau) = (1 + w) \nabla_z \log p(\mathbf{z} | \mathbf{s}) - w \nabla_z \log p(\mathbf{z} | \mathbf{s}^\tau). \quad (12)$$

To analyze this further, we use a first-order Taylor expansion of the degraded score $\nabla_z \log p(\mathbf{z} \mid \mathbf{s}^\tau)$ around \mathbf{s} :

$$\nabla_z \log p(\mathbf{z} \mid \mathbf{s}^\tau) \approx \nabla_z \log p(\mathbf{z} \mid \mathbf{s}) + \frac{\partial}{\partial \mathbf{s}} (\nabla_z \log p(\mathbf{z} \mid \mathbf{s})) \cdot \Delta \mathbf{s}. \quad (13)$$

Substituting this into the update:

$$\nabla_z \log p^{\text{ERG}}(\mathbf{z} \mid \mathbf{s}, \mathbf{s}^\tau) \approx \nabla_z \log p(\mathbf{z} \mid \mathbf{s}) - w \cdot \left(\frac{\partial}{\partial \mathbf{s}} (\nabla_z \log p(\mathbf{z} \mid \mathbf{s})) \cdot \Delta \mathbf{s} \right).$$

Thus, C-ERG introduces a *Jacobian-level product correction*, nudging the sample along directions where the score is most sensitive to perturbations in the conditioning. This helps reinforce fine-grained semantic detail in generation, even when the conditioning signal is noisy or coarsened.

This Jacobian-product reward has a natural interpretation: it reflects how the image-level score $\nabla_z \log p(\mathbf{z} \mid \mathbf{s})$ changes when the conditioning \mathbf{s} is slightly perturbed. This resembles *classifier guidance*, where one steers the generation based on $\nabla_z \log p(\mathbf{y} \mid \mathbf{z})$ —the gradient of a classifier with respect to the image.

Here, the conditioning \mathbf{s} is a continuous vector, and C-ERG guides generation using:

$$\frac{\partial}{\partial \mathbf{s}} (\nabla_z \log p(\mathbf{z} \mid \mathbf{s})) \cdot \Delta \mathbf{s}, \quad (14)$$

which is the direction in image space most sensitive to changes in the semantics of \mathbf{s} . This amounts to amplifying features that are especially responsive to semantic precision.

D Additional Experiments

D.1 C-ERG Impact on diversity

To better understand the effect of c-ERG on diversity, we perform a simple experiment where we track the initial velocity direction during sampling and how it varies when the initial noised input $\mathbf{x}_1 = \boldsymbol{\epsilon} \sim \mathcal{N}(0, I)$ varies.

We examine the variance of the predicted velocity at the start of sampling, conditioning either on the encoding of an empty prompt ϕ_\emptyset or on the encoding of caption using temperature rescaled attention layers ϕ_c^τ . For this we sample $N = 20$ random noise inputs and compare the variance when the inputs vary, the right panel of Figure 5 presents a histogram of the variances where each datapoint corresponds to different spatial/channel location; we refer to it this as *marginal* variance:

$$\begin{cases} \text{Var}_{\substack{\boldsymbol{\epsilon} \sim \mathcal{N}(0, I) \\ c \sim D_c}} [v_\Theta(\boldsymbol{\epsilon}, \phi_\emptyset, t = 0)], \\ \text{Var}_{\substack{\boldsymbol{\epsilon} \sim \mathcal{N}(0, I) \\ c \sim D_c}} [v_\Theta(\boldsymbol{\epsilon}, \phi_c^\tau, t = 0)], \end{cases} \quad (15)$$

where D_c is a dataset of text prompts.

Similarly, in the left panel of Figure 5 we consider the variance of the difference between these velocity estimates and the true conditional one, i.e. when we condition on ϕ_c ; we refer to this as *conditional* variance:

$$\begin{cases} \text{Var}_{\substack{\boldsymbol{\epsilon} \sim \mathcal{N}(0, I) \\ c \sim D_c}} [v_\Theta(\boldsymbol{\epsilon}, \phi_c, t = 0) - v_\Theta(\boldsymbol{\epsilon}, \phi_\emptyset, t = 0)], \\ \text{Var}_{\substack{\boldsymbol{\epsilon} \sim \mathcal{N}(0, I) \\ c \sim D_c}} [v_\Theta(\boldsymbol{\epsilon}, \phi_c, t = 0) - v_\Theta(\boldsymbol{\epsilon}, \phi_c^\tau, t = 0)]. \end{cases} \quad (16)$$

From Figure 5, we observe that standard classifier-free guidance results in a larger range for the conditional variance while the marginal variance is significantly reduced. In contrast, C-ERG exhibits a larger range of marginal variances but a smaller range of conditional variances. Such results indicate that standard classifier-free guidance with high guidance scales operate as an initial condition for the flow ODE (eq. (5)) that has low variance, resulting in exploring only a subset of the solutions of the ODE. Furthermore, as the initial conditional velocity is much larger, it can easily cause overshooting problems as large updates are being taken with a flawed prediction because of imperfections in

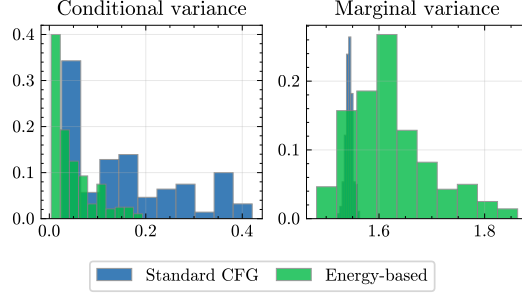


Figure 5: **Effect of C-ERG on initial velocity step.** Standard CFG shows very localized initial marginal variance while the conditional variance is much larger. This means that the negative model predicts an initial velocity that is very similar when varying the initial noise (and prompt in case of C-ERG), and is largely decorrelated from the conditional prediction, leading to a lack of diversity in the generated samples. Conversely, C-ERG results in much higher marginal variance and smaller conditional variance, reducing the error accumulation that can happen at earlier timesteps and leading to better diversity.

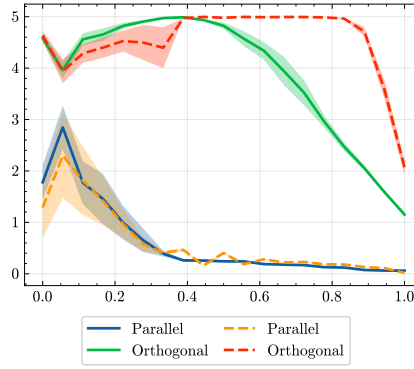


Figure 6: **Orthogonal and parallel differences during sampling.** Solid lines correspond to the parallel and orthogonal differences between the conditional and unconditional prediction of the model under standard CFG while dashed lines correspond to those of I-ERG. The sampling process proceeds from left ($t = 0$) to right ($t = 1$) over the horizontal axis.

the model, while the low marginal variance results in oversaturated colors and simplified image compositions. Similar observations were made by Saharia et al. (2022); Kynkäänniemi et al. (2024); Sadat et al. (2025).

Qualitative results corroborate this, see for example Figure 1 where using standard CFG results in low diversity, saturated images with simple compositions, while our ERG is able to generate more complex scenes that suffer less from the “cartoonish” effect, especially for highly out-of-distribution prompts where the denoiser model is expected to be less accurate.

D.2 I-ERG impact on quality

To better understand the differences between I-ERG and CFG, we consider the difference between the two denoising terms used in CFG and I-ERG, see Equation (16). In particular we decompose the difference into parallel and orthogonal components to the conditional prediction, and track the magnitude of these components throughout the sampling process.

Results in Figure 6 reveal a fundamental difference between I-ERG and CFG. The parallel differences show similar trends between CFG and I-ERG: a peak around $t = 0.075$ followed by a sharp decline towards zero, indicating that for approximately $t \geq 0.4$ the denoising task becomes easy enough that both conditional and unconditional predictions yield similar and correlated results. On the other hand, when examining the orthogonal component of the differences, we see that it keeps a steady norm through sampling while in the case of standard CFG, the norm of the orthogonal difference quickly

decreases as sampling progresses for $t \geq 0.4$, this indicates strong correlations between conditional and unconditional predictions as the denoising task becomes easier and the denoiser needs less to rely on the conditional embeddings.

While [Sadat et al. \(2025\)](#) completely eliminate the parallel component from the difference term, we are able to break such correlations by manipulating the energy landscape of the attention layers of the denoiser. Imposing this divergence between the positive and negative predictors results in semantically meaningful errors for the negative velocity prediction which, when used as a guidance term, improves the low-level details in the resulting image.

Notice that the I-ERG kickoff threshold κ corresponds to the point where the parallel difference between the predicted velocities converges towards 0 while the orthogonal part starts decreasing in a quasi-linear manner, this indicates that the optimal value for κ can be obtained with a simple analysis of the correlation between positive/conditional and negative/unconditional predictions of the model.

D.3 Attention rectification mechanisms

In Table 6, we compare different attention rectification mechanisms from the literature, including energy smoothing (from SEG) ([Hong, 2024](#)), identity mapping (from PAG) ([Ahn et al., 2024](#)) and temperature reduction at the denoiser level as is done in I-ERG. We use a guidance scale of 5.0 and $\tau = 0.2$. Our experiments indicate better performance when using low attention temperature when compared to both energy smoothing and identity mapping in the attention. The most notable difference being in density which is 12 points higher for temperature reduction compared to energy smoothing and 3 points higher than identity mapping. Similarly, temperature reduction achieves the best FID at 14.43 which is 0.6 points lower than energy smoothing and 2.2 points better than identity mapping. Compared to energy smoothing, Identity mapping, as proposed by [Ahn et al. \(2024\)](#), achieves better precision, density, coverage and CLIPScore than energy smoothing but still underperforms when compared to temperature reduction. Similar results can be observed qualitatively in Figure 17, temperature reduction generates images that showcase better realism in terms of the details of the image such as wooden textures on the floor in the first row, and details on the musician’s face and the background in the second row. Similarly, other qualitative results provide evidence supporting this effect, for example in Figure 12 and Figure 18 we observe more detailed backgrounds when using I-ERG than SEG.

Table 6: **Comparison between different mechanisms for attention rectification.** Conducted when using I-ERG.

	FID (\downarrow)	Density (\uparrow)	Coverage (\uparrow)	CLIPScore (\uparrow)
Energy smoothing $\sigma = +\infty$	15.06	102.82	69.45	26.60
Identity mapping	16.62	111.83	70.76	26.42
Temperature reduction	14.43	114.50	71.44	26.71

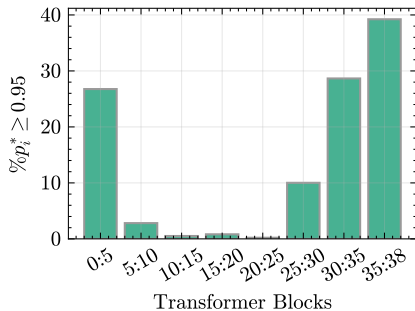


Figure 7: **Percentage of points in the attention matrix that have a high maximum probability.** We observe three different regimes depending on the depth of the block. For blocks [0-5], the maximal association probability is high with more than 25% of tokens having a maximum probability superior to 0.95. For blocks 0-25, the association probability is rarely above 95%. Finally, the proportion of high certainty associations grows back for deeper layers in the model [25-38], reaching almost 40% for the last layers 35 – 38.

D.4 Attention blocks

In order to uncover the role of different attention blocks on the sampling process, we track the number of tokens where the maximum probability in the (non-rectified) attention is superior to a threshold of 0.95, resulting in a near one-to-one matching between the predicted queries and values.

Our results (for $t = 0.3$) are summarized in Figure 7. We find a similar pattern across the different sampling steps. We observe three different regimes through the different layers of the model. While the first and last stage have a reasonably high matching probability, the middle stage has much smaller maximum probabilities and thus combines information from multiple tokens.

With this in mind, we posit that the energy landscape manipulations should only happen in the middle regime, as it would not overly harm the representations of the model. To validate this intuition, we perform a grid search over the range of blocks in which the energy manipulation is performed in Figure 8, we find a similar trend with the emergence of three different regimes, we obtain the best performance when applying the energy guidance on the middle blocks.

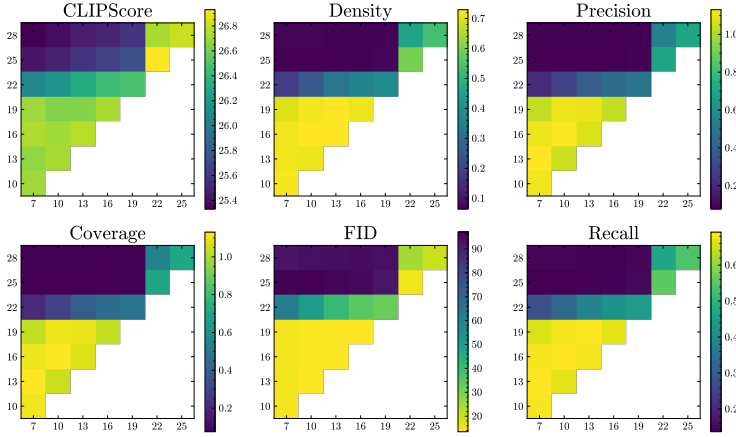


Figure 8: **Choice of rectified attention layers.** Impact of the attention layers where the energy landscape is modified for I-ERG. (x-axis: start block, y-axis: end block). Operating ERG on middle layers seems to perform favorably, while including later layers results in harmful effects.

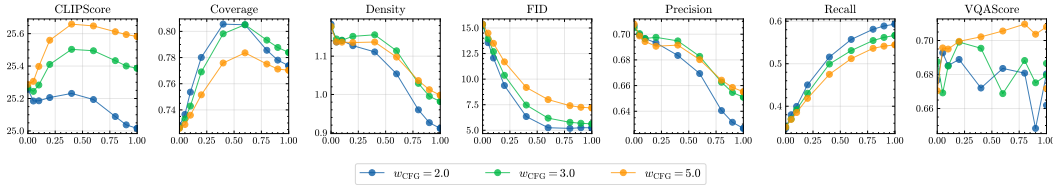


Figure 9: **Influence of I-ERG kickoff threshold κ .** Experiment conducted on our T2I model using Euler sampler with 50 steps. We observe different trends for different metrics, FID and Recall are improved as κ goes to 1, precision and density on the other hand are improved when κ goes to 0, CLIPScore, VQAScore and coverage are optimal for τ in the middle of the range $\kappa \in [0.2, 0.6]$.

D.5 I-ERG kickoff threshold

In Figure 9, we perform a sweep over the kickoff threshold κ for the denoiser-level energy guidance, with the temperature rescaling parameter set as $\tau_i = 0.01$.

For density and precision, we achieve uniform improvements as κ becomes smaller. For CLIPScore, coverage, and VQAScore, we observe better improvements when κ is shifted to the middle of the time range. For recall and FID, we observe consistent degradation as the range of timesteps in which denoiser energy guidance is applied gets larger.

As also reported by Astolfi et al. (2024), there seems to be a natural trade-off between different facets of the generation *i.e* quality, consistency and diversity. While lower thresholds achieve the best precision and density (*quality*), higher thresholds notably improve coverage, recall and FID (*diversity*). On the other hand, CLIPScore and VQA (consistency) are maximized for a mid level threshold. This indicates that κ can be tuned according to the task at hand to achieve suitable results, we find in our experiments that setting $\kappa \in [0.2, 0.4]$ achieves the most balanced performance across different settings.

Table 7: **Sampling parameters for different settings.** We provide the default hyperparameters used in our experiments grouped by model and modality.

Task	Text encoder	τ	l_{\min}^i	l_{\max}^i	τ_i	l_{\min}^c	l_{\max}^c	τ_c	α	γ	K
Text-to-image (MMDiT-B/2)	llama3 T5	0.4	10	17	0.01	0 0	32 24	0.01 0.01	1.0	1.5	1.0
Unconditional (from text2image) (MMDiT-B/2)	—	0.2	10	17	0.01	—	—	—	1.0	1.0	1.0
Class-conditional (DiT-XL/2)	—	0.3	12	15	0.01	—	—	—	1.0	1.0	1.0
Text-to-image (SD3-medium)	OpenCLIP-ViT/G	0.2	6	8	0.01	1	31	0.1	1.0	1.0	1.0
	CLIP-ViT/L					1	10	0.1			
	Flan-T5-XL					1	22	0.1			
Text-to-image (SD3-medium)	OpenCLIP-ViT/G	0.2	10	13	0.01	1	31	0.1	1.0	1.0	1.0
	CLIP-ViT/L					1	12	0.1			
	Flan-T5-XL					1	22	0.1			

Table 8: **FID tuned metrics.** Instead of considering Pareto optimal parameter set, we tune each of the baselines for optimal FID.

Method	CFG	ERG	APG	ERG+APG	PAG	SAG	SEG	CADS	ERG+CADS	AG
FID	5.25	4.93	6.62	5.24	7.00	5.28	6.72	5.12	5.00	7.11

D.6 Optimizing for FID

As shown by Astolfi et al. (2024), FID-optimal checkpoints often lie far from the Pareto front when jointly considering realism, consistency, and diversity. Dhariwal and Nichol (2021) further show that lower guidance scales minimize FID and recall, while higher guidance scales improve perceptual sharpness and structure, measured by IS and precision even though they increase FID.

Consequently, we tune different methods for FID alone, with results reported in Table 8. CFG achieves its best FID at a guidance scale of 1.25, while ERG achieves its optimal FID of 4.93 using a guidance scale of 1.25 and an ERG scale of 2.0. This demonstrates that ERG is capable of outperforming prior guidance methods (e.g., SEG, APG) when optimized for FID alone, in addition to providing better trade-offs under broader evaluation criteria as can be seen in the Pareto fronts in Figure 2 of the main paper.

D.7 Additional Metrics

For better comparability, we extend the results in Table 1 by adding Inception Score (IS), sFID, and Precision & Recall in Table 9. For precision & recall, we observe that similar trends to density and

Table 9: **Additional metrics.** Comparison of guidance methods. Best results are in bold.

Method	CFG	APG	CADS	PAG	SAG	SEG	AutoGuidance	ERG	ERG+APG	ERG+CADS
Precision	65.71	66.34	66.42	66.42	64.97	61.22	61.48	70.92	69.50	72.72
Recall	43.95	45.15	45.75	43.94	47.98	36.88	34.60	41.43	50.25	38.89
sFID	26.53	20.31	18.20	27.21	28.55	33.80	14.75	19.56	9.30	15.32
IS	37.25	43.76	40.75	40.50	38.28	40.64	41.81	43.16	53.47	42.52

coverage with ERG+CADS achieving the best precision (72.72). ERG on its own achieves a precision of 70.92 compared to 65.71 for CFG. Recall is slightly lower when using ERG alone (41.43 vs. 43.95 for CFG), but ERG+APG achieves the best recall of 50.25. For Inception Score, ERG improves over the CFG baseline (43.16 vs. 37.25 for CFG) and achieves the best score when combined with APG (53.47). Similarly, ERG+APG results in significant sFID and Inception score improvement, with ERG alone on par with APG.

E Implementation

In this section, we provide pseudo-code in the style of pytorch to implement ERG.

I-ERG. Changing standard attention mechanism with the energy-based variant defined in Algorithm 1 and applying this for attention blocks between l_{\min} and l_{\max} in the negative/unconditional prediction. Only applied if $t > \kappa$.

C-ERG. For the negative prediction, instead of having the prediction conditioned on empty text tokens, we condition it on tokens obtained by forwarding the text prompts through the text encoder while changing the temperature of the attention in the text encoders between layers l_{\min}^c and l_{\max}^c from β to $\beta \cdot \tau_c$. Applies for all timesteps.

ERG. Consists in applying both I-ERG and C-ERG.

In Algorithm 1, we provide a function that applies the temperature rescaling in the attention layers of text-encoders by utilizing forward hooks on the query mapping. In practice, the temperature rescaling is equivalent to rescaling of the queries prior to the attention operation.

In Algorithm 2, we provide code for the energy-based attention in the case of the mutli-modal transformer block used in Esser et al. (2024).

In Table 7, we provide different hyperparameter sets used for our ERG method under different settings. We consider these values as the defaults used for our experiments unless specified otherwise.

```
def forward_with_temperature(transformer, prompts,
                             tau=0.01, l_min=15, l_max = 20):
    handles = []

    # define forward_hook_function.
    def hook(module, input, output):
        output[:] *= tau
        return output

    # register forward hooks.
    for i in range(l_min, l_max):
        handle = transformer.blocks[i].self_attn.q_proj.register_forward_hook(hook)
        handles.append(handle)

    # encode prompts.
    encoder_hidden_states = transformer.encode(prompts)

    # remove registered_hooks.
    for hook in handles:
        hook.remove()

    return encoder_hidden_states
```

Algorithm 1: **Pseudo-code for temperature scaled encoding of prompts.** Written using Pytorch functionalities.

F Combining different methods

ERG + APG. APG can be seamlessly integrated with ERG by switching the guidance update to the APG update, as both conditional and unconditional predictions can be converted into clean latent estimates, the algorithm operates as described by Sadat et al. (2025). In the case of rectified flows, such a conversion is easily obtained as $\hat{\mathbf{x}}_0 = \mathbf{x}_t + (1 - t) \cdot \hat{\mathbf{v}}_{\Theta}(\mathbf{x}_t, \phi, t)$. After the guidance update, the clean image estimate is projected back into velocity prediction by inverting the formula: $\hat{\mathbf{v}}_{\Theta}(\mathbf{x}_t, \phi, t) = (\hat{\mathbf{x}}_0 - \mathbf{x}_t)/(1 - t)$.

ERG + CADs. CADs is also straightforward to integrate with ERG, for this we switch the text encoder hidden states in the conditional/positive prediction with the interpolation between the tokens and Gaussian noise as described by Sadat et al. (2024).

```

def multistep_attention(q,k,v,step_size=1.0,steps=1, gamma=1.0):
    q_new = q.clone()
    for i in range(steps):
        att = scaled_dot_product_attention(q_new,k,v)
        q_new = q_new - step_size * (q_new - gamma * att)
    return q_new

def energy_based_attention(self, x, rope_freqs, num_img_tokens, tau_i):
    B, N, C = x.shape
    # B: batch size — N: Number of tokens — C: Hidden dimension.
    s1 = num_img_tokens

    q = self.q_linear(x).reshape(
        B, N, num_heads, C // num_heads
    ).permute(0, 2, 1, 3)
    kv = kv_linear(x).reshape(
        B, N, 2, num_heads, C // num_heads
    ).permute(2, 0, 3, 1, 4)

    k, v = kv.unbind(0)
    q = self.q_norm(q)
    k = self.k_norm(k)

    # Optionally apply rotary positional embeddings.
    qi, qc = q[:, :, :s1], q[:, :, s1:]
    ki, kc = k[:, :, :s1], k[:, :, s1:]
    qi, ki = apply_rotary_emb(qi, ki, rope_freqs=rope_freqs)

    # Entropy rectification in the image tokens.
    if tau_i > 0:
        qi[:, :] = qi[:, :] * tau_i

    q = torch.cat([qi, qc], dim=2)
    k = torch.cat([ki, kc], dim=2)

    if not self.use_e_att:
        x = scaled_dot_product_attention(q, k, v)
    else:
        x = multistep_attention(
            q,
            k,
            v,
            lr=self.step_size,
            steps=self.num_steps,
            gamma=self.potential_w,
        )

    x = x.transpose(1, 2).reshape(B, N, C)
    x = proj(x)
    return x

```

Algorithm 2: **Pseudo-code for temperature scaled encoding of prompts.** The hidden state x consists of image tokens concatenated with text/class tokens.

G Reimplementation

We implement our method on the open-source Stable Diffusion 3 model, we use the `sd3-medium` from `diffusers` library. We refer to Table 7 for details about the choice of hyperparameters. Qualitative examples on Figure 19 show significant improvements in terms of image quality. In Table 10, we report quantitative results on this model on COCO-5k benchmark. For this experiment we use the standard setup with a guidance scale of 5.0 and Euler sampler with 28 steps. We observe significant improvements for all metrics when using ERG compared to standard classifier-free guidance. More precisely, we observe significant boosts in FID -10.03 pts, density $+73.56$, coverage $+13.34$ and CLIPScore $+3.0$.

We implement ERG on different architectures beyond the standard DiT. Specifically, we provide additional (I-)ERG results for SDXL, PixArt-alpha and EDM2 (autoguidance). SDXL and EDM2 use U-Net architectures while PixArt-alpha uses local window attention. For SDXL and Pixart-alpha, we use the opensource version from Hugging Face alongside the EvalGIM evaluation framework. For EDM2 and DiT, we use the evaluation setups from their respective open-source implementations.

Table 10: **Effect of ERG on Stable Diffusion 3 models.** We compare standard classifier-free guidance with our method on COCO-5K benchmark. We observe significant improvements across reported metrics.

		FID (\downarrow)	Density (\uparrow)	Coverage (\uparrow)	CLIPScore (\uparrow)
SD3.0	CFG	35.85	41.11	10.95	24.71
	ERG	25.82	114.67	24.29	27.71
SD3.5	CFG	34.33	37.94	11.76	28.66
	ERG	23.81	118.59	25.26	29.61
PixART-alpha	CFG	28.64	52.98	13.46	26.03
	ERG	27.75	70.61	16.57	26.61
SDXL	CFG	21.65	80.93	20.00	29.54
	ERG	20.94	84.05	20.21	29.97

On EDM2/Autoguidance. As reported in Table 11, ERG improves over baseline EDM2 and obtains similar FID and slightly better FID_dinov2 than the autoguidance results, without need for additional models.

Table 11: **EDM and Autoguidance comparison**

Model/Metric	FID (\downarrow)	FID_dinov2 (\downarrow)
EDM2-IN-XXL-512	2.00	59.34
EDM2-IN-XXL-512 + ERG	1.35	31.32
EDM2-IN-XXL-512 + Autoguidance	1.33	31.56

On open source DiT. For completeness, we provide comparative results of our method applied to the official opensource implementation of DiT (Peebles and Xie, 2023). We re-ran the experiment using the guided-diffusion library to match exactly the evaluation setup from the DiT paper for 10 different random seeds. The results of the rectified experiment are reported in Table 12 and Table 13.

Table 12: **Open source DiT results.** Compared with the standard DiT.

Model/Metric	FID	sFID	Precision	Recall	IS
DiT-XL/2-256	2.38	4.55	82.77	57.94	277.96
DiT-XL/2-256 + ERG	2.15	4.32	86.90	60.03	295.03

Table 13: **Statistical analysis on FID.** We report aggregate statistics over 10 different random seeds, minimum and maximum values are reported between brackets.

	IS	FID	sFID	Precision	Recall
Baseline	278.67 \pm 4.93 [270.67-273.39]	2.38 \pm 0.05 [2.31-2.48]	4.55 \pm 0.12 [4.41-4.55]	82.74 \pm 0.16 [82.48-82.75]	57.96 \pm 0.41 [57.34-58.50]
I-ERG	293.79 \pm 2.06 [289.18-297.84]	2.15 \pm 0.03 [2.09-2.18]	4.28 \pm 0.08 [4.26-4.33]	86.94 \pm 0.09 [86.82-87.06]	59.98 \pm 0.42 [59.21-60.64]

For the baseline, we obtain an average FID of 2.38 which is higher than the one reported in the paper but is plausible given the variance of the results. Compared to the baseline, I-ERG still achieves a better FID (2.15 vs. 2.38), and given the statistics, also yields smaller variance on all metrics other than recall, showing more stable performance while varying random seeds.

H Assets

In Table 14 we provide the links to the datasets and models used in our work and their licensing.

I Additional qualitative results

We provide additional qualitative examples of our model under different settings.

In Figure 10, we provide a comparison between classifier-free guidance and Entropy Rectifying Guidance on three different prompts with varying levels of detail, we showcase the conditional diversity of the methods by providing samples with 5 different random seeds for each prompt. For longer prompts (top row), we observe similar consistency but higher diversity in terms of background colors, the dog’s physical traits etc. Similarly, the second row shows increased diversity with ERG in terms of colors, textures and the drawing that the cat is holding. For very short prompts (third row),

Table 14: **Reference for the different assets used in our work.**

COCO'14	https://www.cocodataset.org
ImageNet	https://www.image-net.org
CC12M	https://github.com/google-research-datasets/conceptual-12m
YFCC100M	https://www.multimediacommons.org
Llama3-8b	https://huggingface.co/meta-llama/Meta-Llama-3-8B
Flan-T5-XL	https://huggingface.co/google/flan-t5-xl
Stable Diffusion 3	https://huggingface.co/stabilityai/stable-diffusion-3-medium
EvalGIM	https://github.com/facebookresearch/EvalGIM

we observe certain redundancies in the standard CFG generations while ERG is able to generate high quality samples showcasing higher diversity (depicting the horse against different backgrounds, different points of view, different time of day etc.).

In Figure 11, we provide a comparison between classifier-free guidance and I-ERG showcasing the improvements in image quality when generating samples with the same random seed. By setting $\tau_i = 0.2$, the image semantics are not steered excessively compared to the CFG generations, resulting in comparable generations. We observe noticeable improvements in terms of high level details in the images when using I-ERG.

In Figure 12, we provide comparisons between vanilla generation (no guidance), our implementations of SAG, PAG, SEG, and I-ERG for unconditional image generation. While PAG*, and SEG* are also able to improve image quality to satisfying levels, I-ERG provides an additional level of detail that is not present in the other methods. See for example the table texture on second column, background in the third and fifth column, water texture in fourth column and grass texture in the fifth column.

In Figure 13, we provide qualitative comparisons between different methods for text-to-image generation using a single prompt and multiple seeds. We compare our method with CFG, CADs and APG. We also provide samples generated when combining our method with either APG or CADs. In each column the same random seed is used for all generations. We observe that both CFG and CADs tend to generate images that have a cartoonish style, APG on the other hand generates images that are more realistic. ERG alone significantly boosts image realism and results in less saturated images (pure white/black backgrounds for example). Mixing, ERG with either CADs or APG results in significant improvements in both image quality and diversity.

In Figure 14, we provide a qualitative comparison between APG and APG + ERG, similar to Figure 1, we observe improvements in image quality and diversity. Also noticeable is a drift from unrealistic image styles (cartoonish) towards more realistic as can be seen in the astronaut and panda examples.

In Figure 15, we showcase the effect of the image kickoff threshold κ and denoiser attention temperature τ_i by interpolating both parameters in an unconditional generation experiment. We observe that lowering the attention temperature $\tau_i \rightarrow 0$ results in improved image quality. Similarly, lowering the image kickoff threshold κ further improves image quality but steers the semantics of the image away from the vanilla generation ($\kappa = 1.0, \tau_i = 1$). However, as the image level threshold gets smaller $\kappa \rightarrow 0$, the semantics of the original image diverge from the vanilla generations, resulting in images with largely different semantics. We find $\kappa \in [0.2, 0.4]$ and $\tau_i = 0.01$ to work well in most cases.

In Figure 16, we illustrate the effect of applying I-ERG on different layers of the denoiser. This experiment is conducted for class-conditional ImageNet-1k generations with our DiT-XL/2 model with 28 layers. We apply I-ERG to a different range of layers where three layers are involved each time. Applying I-ERG to either very early or late layers results in similar effects, with noticeable over-saturation and unnatural patterns that are overly simplistic as can be seen in the dog's fur. On the other hand, applying I-ERG to middle layers results in improved image quality and sharper details. We find $l_{\min} = 12, l_{\max} = 15$ to work well and use it for the rest of the experiments in this setup.

In Figure 18, we provide qualitative results comparing I-ERG with SEG and CFG for class-conditional generation on ImageNet at 512 resolution, using a DiT-XL/2 model. Similarly to the unconditional and text-to-image cases, we find I-ERG to achieve higher image quality when compared CFG and our reimplementations of other guidance methods.

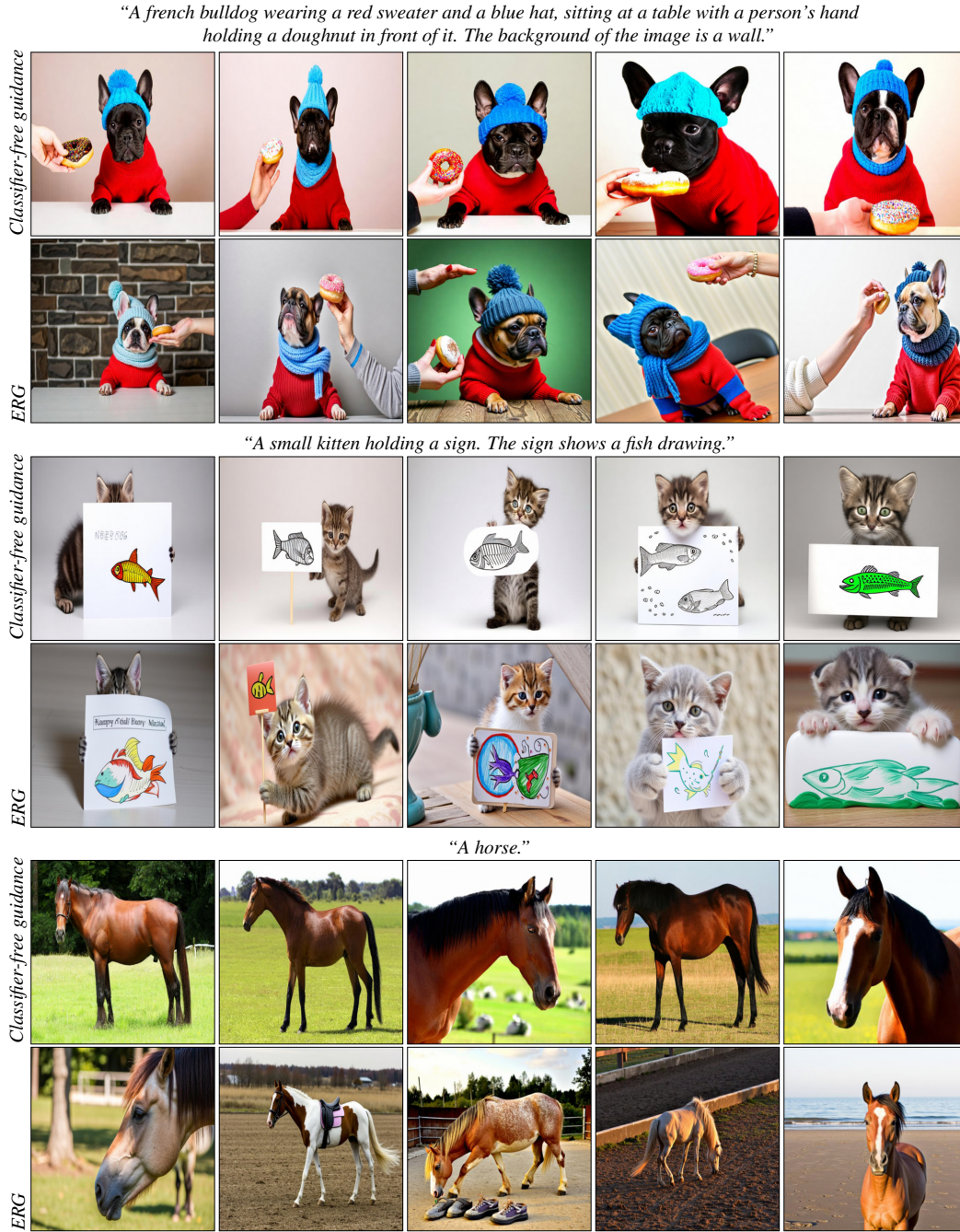


Figure 10: **Qualitative comparison standard classifier-free guidance (top) and Entropy Rectifying Guidance (bottom).** Each column represents images generated using the same random seed. Images generated with scaled temperature show more complex textures and variations than the standard guidance.



Figure 11: **Influence of I-ERG on boosting image quality.** Top row: images generated with standard guidance. Bottom row: images generated with $\alpha = 0.01$, a guidance scale of $w_{CFG} = 3.0$ and $\tau = 0.2$ order to match the image semantic in the comparison.

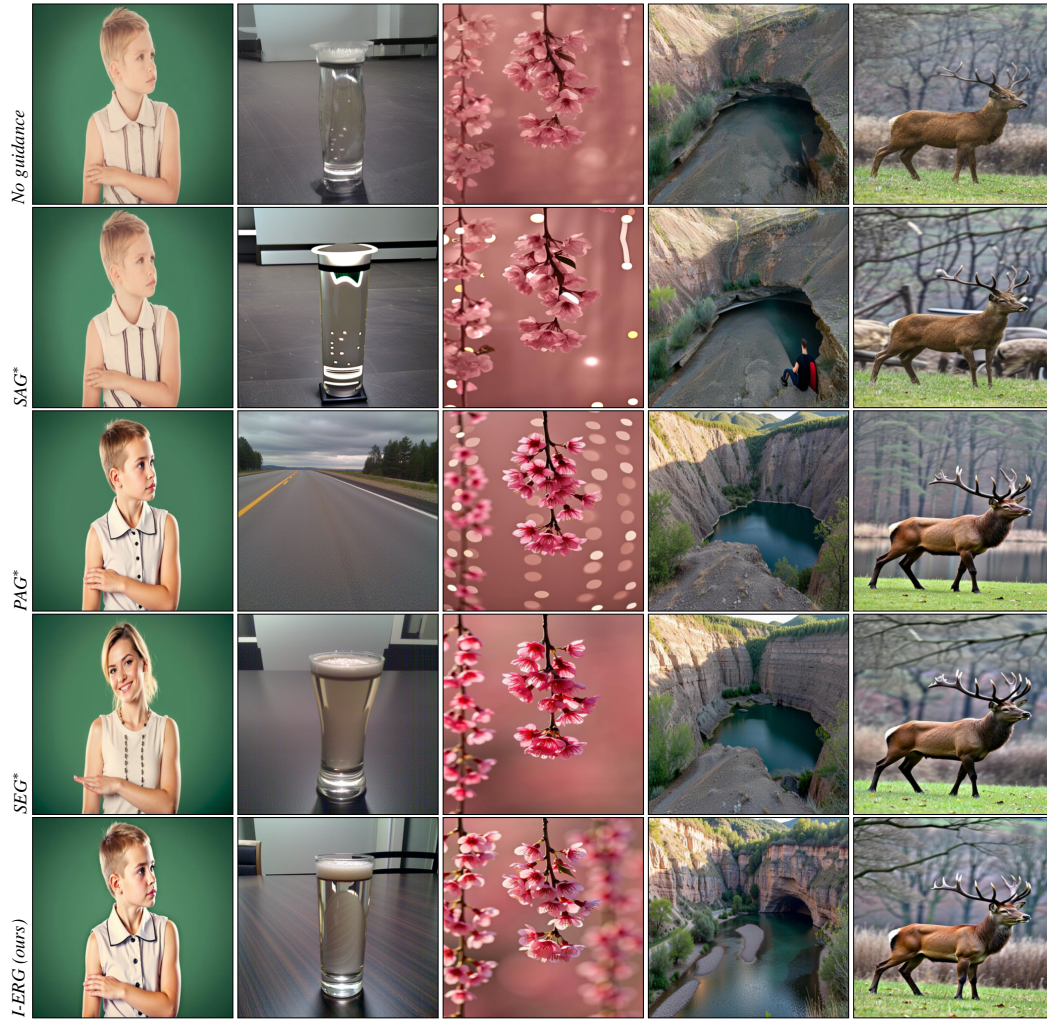


Figure 12: **Comparing different methods for unconditional image generation.** Images are sampled from the text-to-image model while conditioning on an empty prompt, using a guidance scale of 3.0 and Euler sampler with 50 sampling steps.



Figure 13: **Qualitative comparison between different methods.** We provide a detailed qualitative comparison between different methods, we use the “A lion with sunglasses and a suit, seated in a sofa, reading the newspaper.” and sample images with 5 different random seeds, each corresponding to a column in the figure.

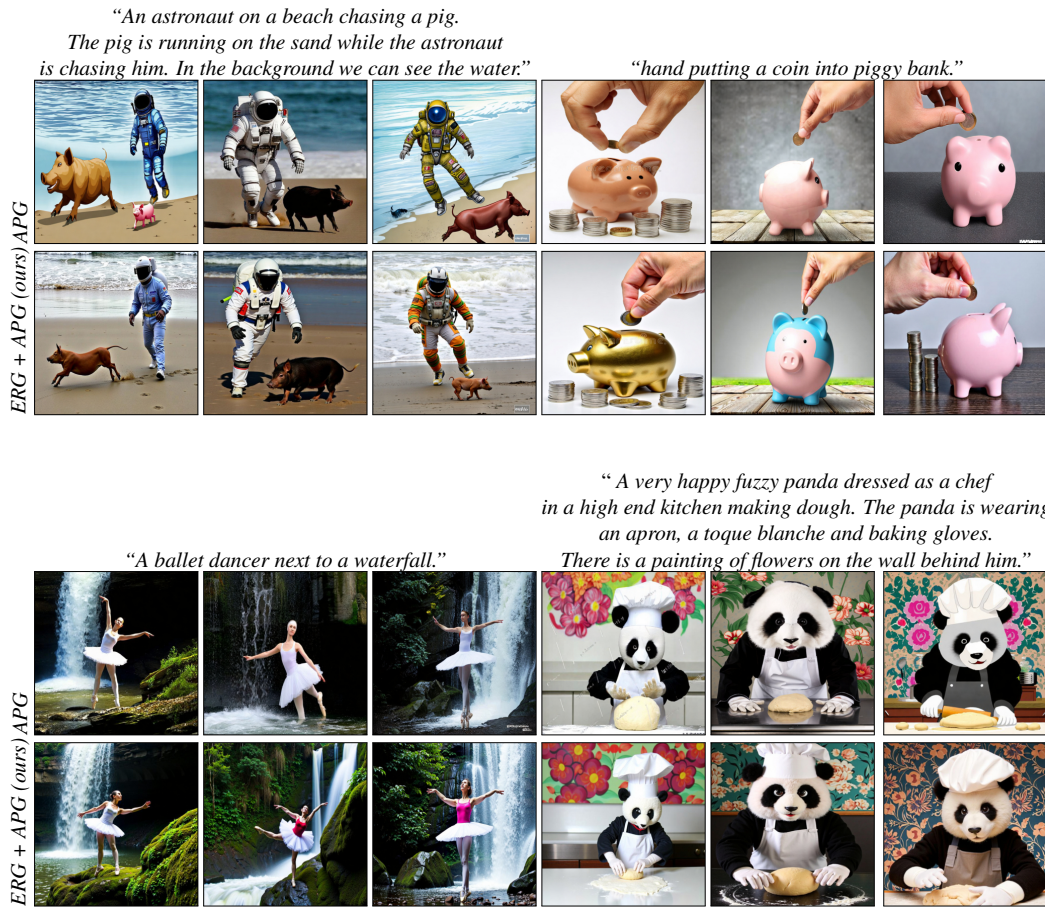


Figure 14: Qualitative comparison showcasing the effect of Entropy Rectifying Guidance (ERG) on generation quality when using APG.



Figure 15: **Effect of I-ERG main parameters.** Interpolating the image kickoff threshold κ on the vertical axis and denoiser attention temperature τ_i on the horizontal axis showcases the effect on image quality and global semantics of the generated image.

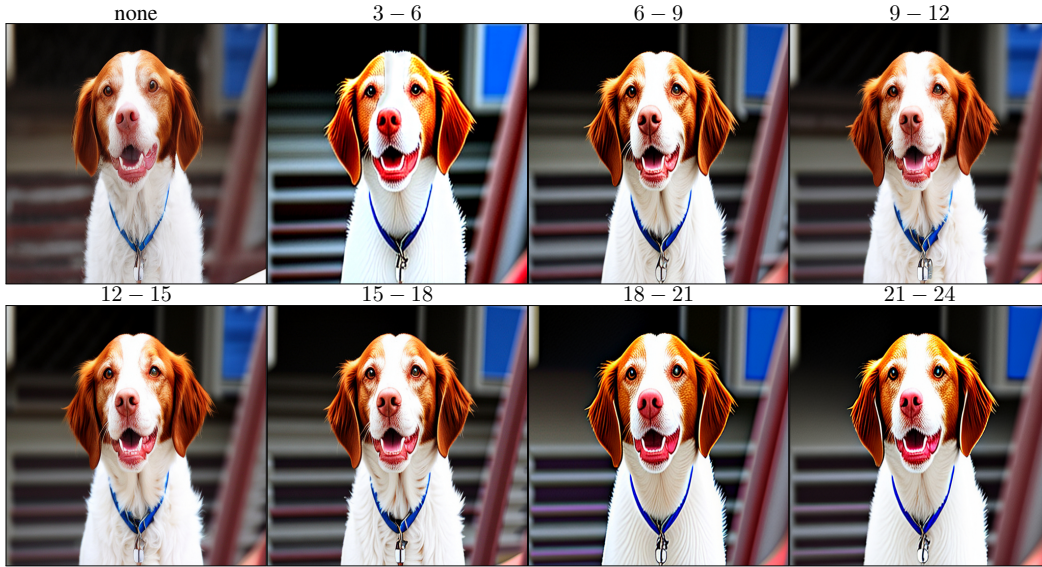


Figure 16: **Effect of I-ERG main parameters.** We show the effect of applying I-ERG on different layers of the transformer l_{\min}^i, l_{\max}^i . Experiment conducted on DiT-XL/2 model, using a guidance scale of 5.0.



Figure 17: **Comparing attention rectification mechanisms.** Compared to attention smoothing and identity mapping, temperature rescaling better models high-level details in the image such as textures of the tiling and small buildings in the distance. Temperature rescaling also showcases better structural coherence when compared with different methods.

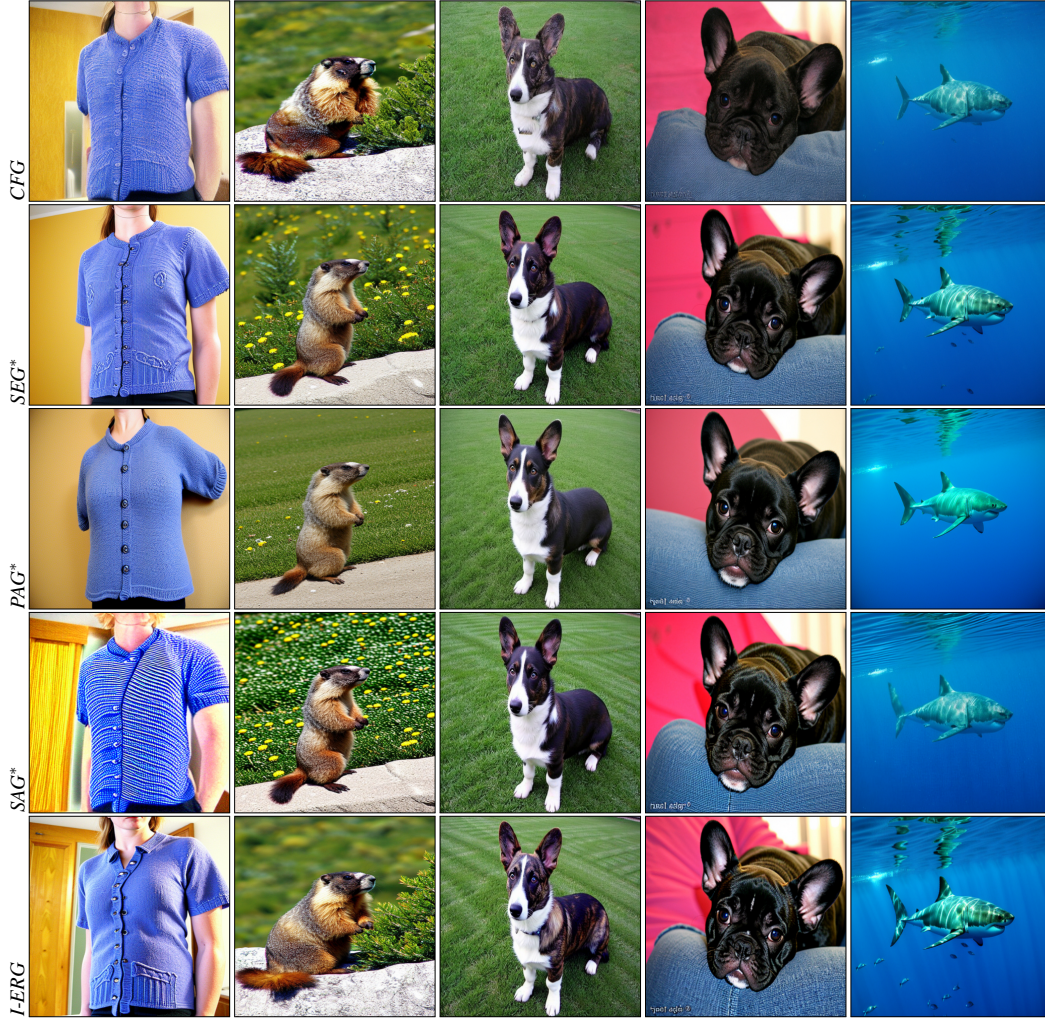


Figure 18: **Qualitative comparison on ImageNet-1k@512.** Samples from the same column are generated using the same random seed, we compare ERG with CFG and SEG. Our method produces images of better quality. Images generated using Euler sampler with 50 steps.

An elephant crossing a river.



A group of supporters wearing football jerseys cheering for their team after a victory.



A child wearing an apron and sunglasses posing in front of a blackboard with heart drawings and 'ERG' written in it.



A maltese dog getting a haircut by a hairdresser



Figure 19: **Qualitative comparison using Stable Diffusion 3.** For each prompt we use the same seed to sample images from sd3-medium (top rows) and sd3.5-large (bottom rows) with standard CFG (top) and I-ERG (bottom). We observe better image details using I-ERG.