Boosting Latent Diffusion with Flow Matching

Johannes S. Fischer* Ming Gui* Pingchuan Ma*
Nick Stracke Stefan A. Baumann Björn Ommer
LMU Munich, MCML

Abstract

Recently, there has been tremendous progress in visual synthesis and the underlying generative models. Here, diffusion models (DMs) stand out particularly, but lately, flow matching (FM) has also garnered considerable interest. While DMs excel in providing diverse images, they suffer from long training and slow generation. With latent diffusion, these issues are only partially alleviated. Conversely, FM offers faster training and inference but exhibits less diversity in synthesis.

We demonstrate that introducing FM between the Diffusion model and the convolutional decoder offers high-resolution image synthesis with reduced computational cost and model size. Diffusion can then efficiently provide the necessary generation diversity. FM compensates for the lower resolution, mapping the small latent space to a high-dimensional one. Subsequently, the convolutional decoder of the LDM maps these latents to high-resolution images. By combining the diversity of DMs, the efficiency of FMs, and the effectiveness of convolutional decoders, we achieve state-of-the-art high-resolution image synthesis at 1024^2 *with minimal computational cost. Importantly, our approach is orthogonal to recent approximation and speed-up strategies for the underlying DMs, making it easily integrable into various DM frameworks.*

1. Introduction

Visual synthesis has recently witnessed unprecedented progress and popularity in computer vision and beyond. Various generative models have been proposed to address the diverse challenges in this field [61], including sample diversity, quality, resolution, and training, and test speed. Among these approaches, diffusion models (DMs) [46, 47, 49] currently rank among the most popular and highest quality, defining the state of the art in numerous synthesis applications. While DMs excel in sample quality and diversity, they face challenges in high-resolution synthesis, slow sampling speed, and a substantial memory footprint.

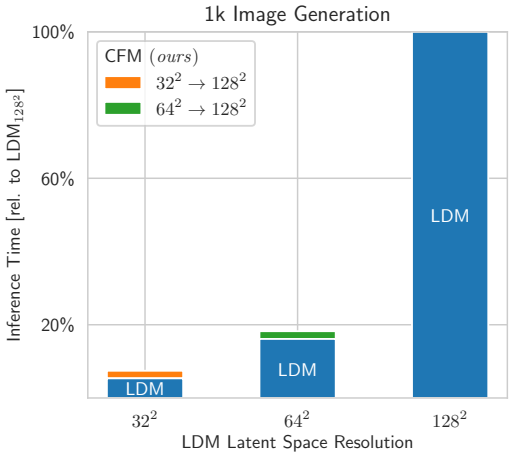

Figure 1. Comparison between 1k image synthesis using different architectures. We utilize SD V1.5 as our base model for LDM and scale the attention accordingly based on [21]. LDM's inference time grows quadratically with higher resolutions, making real-time inference nearly impractical at a 128^2 resolution latent space. In contrast, the integration of Coupling Flow Matching model (CFM) with 50 function evaluations exhibits consistently faster inference, highlighting its efficiency in handling high-resolution image synthesis.

Lately, numerous efficiency improvements to DMs have been proposed [10, 45, 56], but the most popular remedy has been the introduction of Latent Diffusion Models (LDMs) [47]. Operating only in a compact latent space, LDMs combine the strengths of DMs with the efficiency of a convolutional encoder-decoder that translates the latents back into pixel space. However, Rombach et al. [47] also showed that an excessively strong first-stage compression leads to information loss, limiting generation quality. Efforts have been made to expand the latent space [43] or stack a series of different DMs, each specializing in different resolutions [19, 49], but these approaches are still computationally costly, especially when synthesizing high-resolution

*Equal Contribution
Project Page: https://github.com/CompVis/fm-boosting

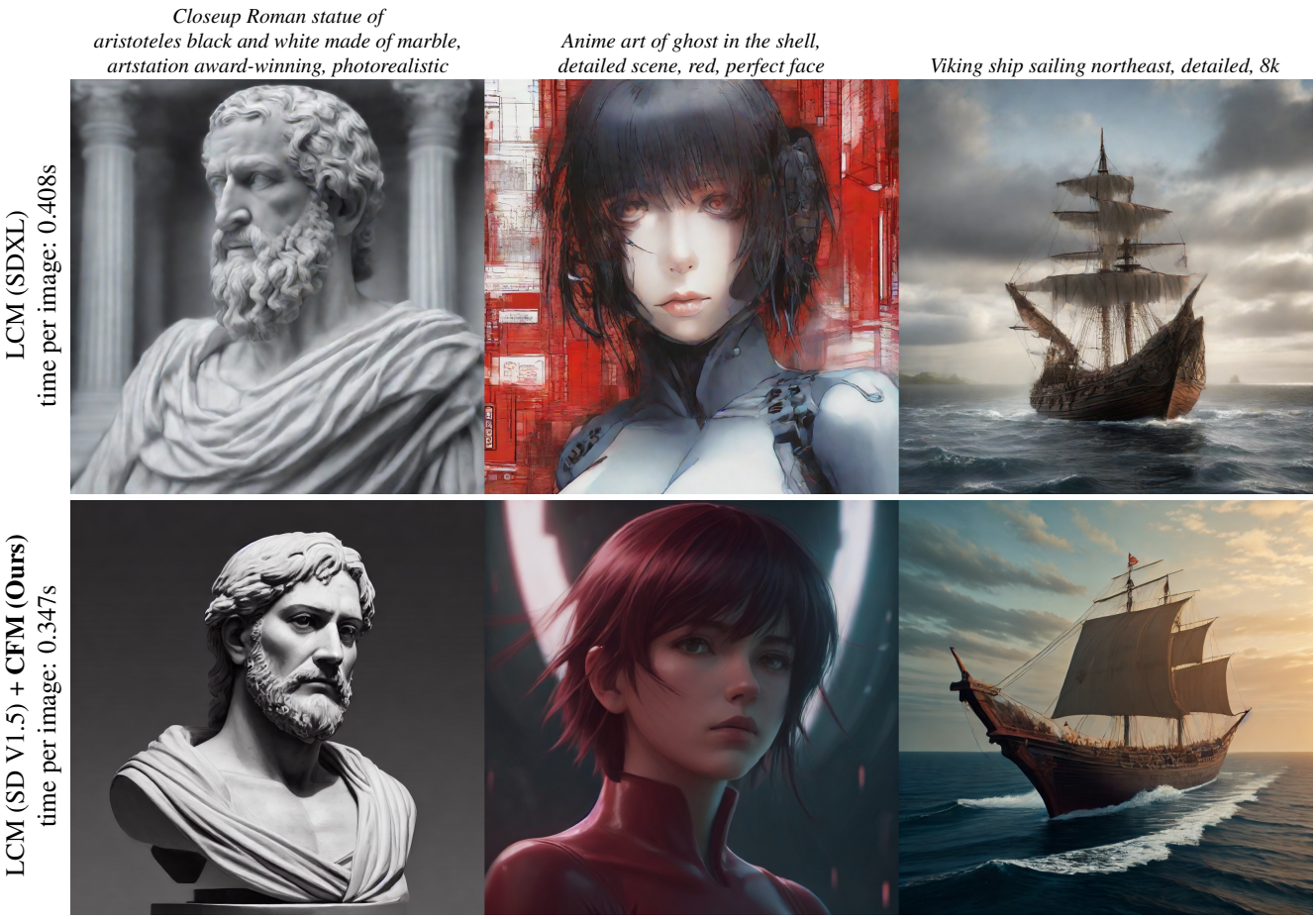

Figure 2. Samples synthesized in 1024^2 px. We elevate DMs and similar architectures to a higher-resolution domain, achieving exceptionally rapid processing speeds. We leverage the Latent Consistency Models (LCM) [38], distilled from SD V1.5 [47] and SDXL [43] respectively. To achieve the same resolution as LCM(SDXL), we boost LCM(SD V1.5) with our general Coupling Flow Matching (CFM) model. This yields a further speedup in the synthesis process and enables the generation of high-resolution images of high fidelity in average 0.347 seconds. The LCM(SDXL) model fails to produces competitive results within this shortened timeframe, highlighting the effectiveness of our approach in achieving both speed and quality in image synthesis.

images.

The inherent stochasticity of DMs is key to their proficiency in generating diverse images. In the later stages of DM inference, as the global structure of the image has already been generated, the advantages of stochasticity diminish. Instead, the computational overhead due to the less efficient stochastic diffusion trajectories becomes a burden rather than helping in up-sampling to and improving higher resolution images [7]. At this stage, converse characteristics become beneficial: reduced diversity and a short and straight trajectory towards the high-resolution latent space of the decoder. These goals align precisely with the strengths of Flow Matching (FM) [31], another emerging family of generative models currently gaining significant attention. In contrast to DMs, Flow Matching enables the modelling of an optimal transport conditional probability path between two distributions that is significantly straighter than those achieved by DMs, making it more robust, and efficient to train. The deterministic nature of Flow Matching models also allows the utilization of off-the-shelf ODE solvers, which are more efficient to sample from and can further accelerate inference.

We leverage the complementary strengths of DMs, FMs, and VAEs: the diversity of stochastic DMs, the speed of Flow Matching in training and inference stages, and the efficiency of a convolutional decoder when mapping latents into pixel space. This synergy results in a small diffusion model that excels in generating diverse samples at a low resolution. Flow Matching then takes a direct path from this lower-resolution representation to a higher-resolution latent, which is subsequently translated into a high-resolution image by a convolutional decoder. Moreover, the Flow

Matching model can establish data-dependent couplings with the synthesized information from the DM, which automatically and inherently forms optimal transport paths from the noise to the data samples in the Flow Matching model [5, 59].

Note that our work is complementary to recent work on sampling acceleration of diffusion models like DDIM [56], DPM-Solver [36], and LCM-LoRA [37, 38]. Our approach can be directly integrated into any existing diffusion model architecture to efficiently increase the final output resolution.

2. Related Work

Diffusion Models Diffusion models [18, 55, 57] have shown broad applications in computer vision, spanning image [47], audio [32], and video [8, 20]. Albeit with high fidelity in generation, they do so at the cost of sampling speed compared to alternatives like Generative Adversarial Networks [15, 22, 24]. Hence, several works propose more efficient sampling techniques for diffusion models, including distillation [39, 51, 58], noise schedule design [26, 42, 44], and training-free sampling [25, 33, 35, 56]. Nonetheless, it is important to highlight that existing methods have not fully addressed the challenge imposed by the strong curvature in the sampling trajectory, which limits sampling step sizes and necessitates the utilization of intricately tuned solvers, making sampling costly.

Flow Matching-based Generative Models A recent competitor, known as Flow Matching [4, 31, 34, 40], has gained prominence for its ability to maintain straight trajectories during generation by modeling the synthesis process using an optimal transport conditional probability path with ordinary differential equations (ODE), positioning it as an apt alternative for addressing trajectory straightness-related issues encountered in diffusion models. The versatility of Flow Matching has been showcased across various domains, including image [12, 31], video [6], audio [27]. This underscores its capacity to address the inherent trajectory challenges associated with diffusion models, mitigating the limitations of slow sampling in the current generation based on diffusion models. Considerable effort has been directed towards optimizing transport within Flow Matching models [34, 59], which contributes to enhanced training stability and accelerated inference speed by making the trajectories even straighter and thus enabling larger sampling step sizes. However, the efficacy of Flow Matching in generation capabilities presently does not parallel that of diffusion models [12, 31].

This may be attributed to the presence of random noise within the training process as opposed to diffusion models, where it increases the stochasticity and flexibility of the generation process. In this paper, we remedy this limitation by incorporating a small diffusion model for synthesis quality.

Image Super-Resolution Image super-resolution (SR) is a fundamental problem in computer vision. Prominent methodologies include GANs [22, 28, 60, 64], diffusion models [29, 50, 63] and Flow Matching methods [5, 31].

Our methodology adopts the Flow Matching approaches, leveraging it's objective to achieve faster training and inference compared to diffusion models. We take inspiration from latent diffusion models [47] and transition the training to the latent space, which further enhances computational efficiency. This enables the synthesis of images with significantly higher resolution, thereby advancing the capacity for image generation in terms of both speed and output size.

3. Method

We speed up and increase the resolution of existing LDMs by integrating Flow Matching in the latent space. The proposed architecture should not be limited to unconditional image synthesis but also be applicable to text-to-image synthesis [41, 46, 47, 49] and Diffusion models with other conditioning including depth maps, canny edges, etc. [14, 30, 65].

The main challenge is not a deficiency in diversity within the Diffusion model; rather it is the slow convergence of the training procedure, the huge memory demand, and the slow inference [43, 49, 62]. While there are substantial efforts to accelerate inference speed of DMs either by distillation techniques [39], or by an ODE approximation at inference [35, 36, 56], we argue that we can achieve faster training and inference speed by training with an ODE assumption [31]. Flows characterized by straight paths without Wiener process inherently incur minimal time-discretization errors during numerical simulation [34] and can be simulated with only few ODE solver steps.

Outline Our solution to high-resolution image generation involves a small Diffusion model and a dedicated Flow Matching model. In Sec. 3.1, we explore the intricacies of the Diffusion model. Its reduced size can be strategically chosen to strike a balance between model complexity and performance, ensuring optimal proficiency in capturing complex data distributions. The Flow Matching model specializes in high-resolution image generation, which we will cover in Sec. 3.2. Finally, the combination of both methods is detailed in Sec. 3.3, illustrating the combination of the strengths of the Diffusion and Flow Matching models for comprehensive high-resolution image generation.

3.1. Diffusion Models

Diffusion models are likelihood-based probabilistic models that learn a data distribution $p(x)$ by reversing a fixed for-

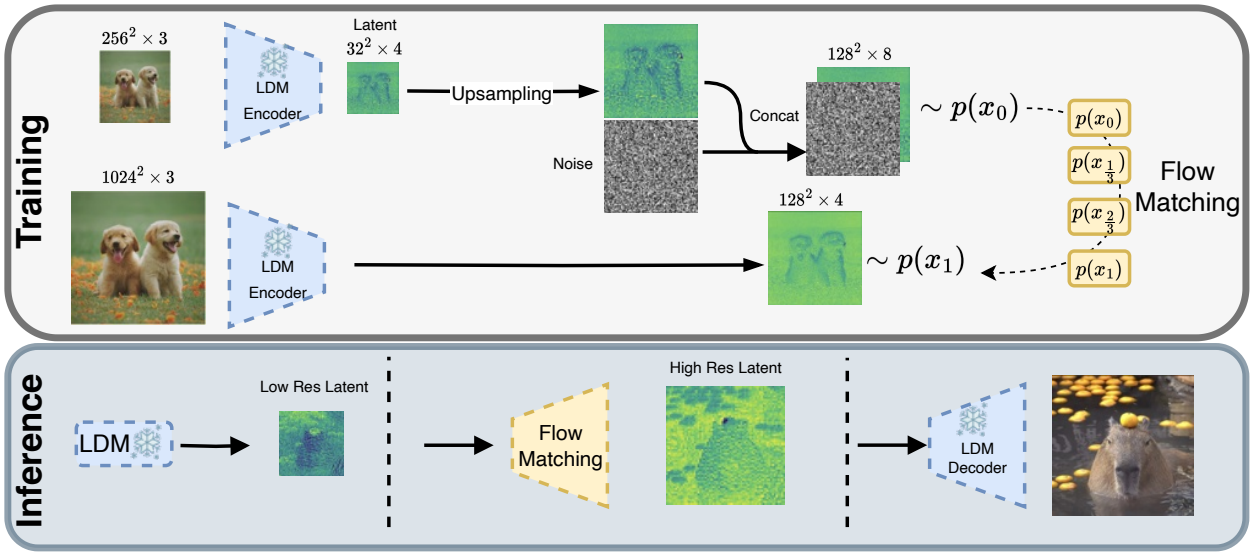

Figure 3. Approach overview. *Top*. During training we feed both a low- and a high-resolution image through the pre-trained Latent Diffusion Encoder to obtain a low- and a high-resolution latent code, respectively. In order to match the dimensionality we up-sample the low-resolution image with either bi-linear or nearest neighbor up-sampling. Depending on the model, we concatenate Gaussian noise and perform flow matching for a step $t \in [0, 1]$ between the up-sampled low-resolution latent and the high-resolution latent. *Bottom*. During inference we can take any Latent Diffusion model that uses the same Encoder-Decoder architecture, generate the low-resolution latent, and then use the flow matching model to synthesize the higher dimensional latent code. Finally, the pre-trained decoder projects the latent code back to pixel space.

ward noising process. Starting with data samples x_0, a fixed Markov chain adds noise to it, creating increasingly noisy versions x_t,

$$q(x_t \mid x_0) = \mathcal{N}(\alpha_t x_0, \sigma_t^2 \mathbf{I}), \tag{1}$$

where α_t and σ_t^2 are positive scalar-valued functions of timestep $t \in [0, T]$ that determine the signal-to-noise ratio of the noising process. The signal of x_0 is completely destroyed at timestep T so that x_T follows a standard Gaussian distribution with the same dimensionality as x_0. The training objective of Diffusion models is a reweighted variant of the variational lower bound on $p(x)$ [18],

$$L_{\text{simple}}(\theta) := \mathbb{E}_{t, x_0, \epsilon \sim \mathcal{N}(0, \mathbf{I})} \left[\|\epsilon - \hat{\epsilon}_\theta(x_t, t)\|^2 \right], \tag{2}$$

where $\hat{\epsilon}_\theta$ is a function approximator that tries to model the reverse process noise at each intermediate step of the Markov chain, and t is uniformly sampled in the range $[1, T]$ during training. During inference, $\hat{\epsilon}_\theta$ is used to iteratively denoise a Gaussian for T steps until it finally arrives at a data sample $x_0 \sim p(x)$.

The stochasticity inherent in Diffusion models enables them to effectively approximate the data manifold, even for high-dimensional complex data such as images [42, 49] and videos [8, 20, 53]. However, to arrive at this high data variety, Diffusion models usually require a large number of training iterations on vast amounts of training data, making them inefficient for high-resolution images. To mitigate the problem and to foster high-resolution image synthesis, Rombach et al. [47] propose *Latent Diffusion Models* (LDMs). With a pre-trained Kullback-Leibler-regularized autoencoder, they perceptually compress images to a lower-dimensional latent space and perform Diffusion training in this space. Assuming an encoder $\mathcal{E}$, the objective from Eq. (2) can then be expressed as

$$L_{\text{LDM}}(\theta) := \mathbb{E}_{t, \mathcal{E}(x_0), \epsilon \sim \mathcal{N}(0, \mathbf{I})} \left[\|\epsilon - \hat{\epsilon}_\theta(z_t, t)\|^2 \right], \tag{3}$$

where z_t refers to the encoded image noised to timestep t with the forward noising process described in Eq. (1). The decoder $\mathcal{D}$ thereafter decompresses the latent z_0 back into pixel space.

LDMs exhibit multiple desirable properties: the stochastic formulation of Diffusion models allows us to model a large data diversity, generating an image's global structure and semantics. The compression into latent space additionally reduces computational demands during training, whereas during inference, we can utilize the decoder to recover high-frequency details and the higher-resolution image from the generated latent. In combination, both provide for great diversity and fast training and inference. Nonetheless, compression with the autoencoder is limited so that stronger compression deteriorates quality [47].

From LDM to FM-LDM Hence, while the stochastic Diffusion model allows diverse sampling and the first stage compresses from pixel space into the latent space, we seek an orthogonal approach to model the straight trajectory from low-resolution to higher-resolution representations. Given the already generated semantics of an image from the Diffusion model, we claim that there is no need to utilize further stochastic processes to map to a higher dimensional space. Assuming a direct mapping, we propose modeling this part as a straight trajectory, which allows more efficient training and faster inference. Concretely, by letting Diffusion models focus only on semantic information, we can reduce their size and enhance the generation process with simpler, more lightweight models. Recently, the formulation of generative processes as optimal transport conditional probability paths has gained much attraction [4, 31, 59], perfectly suiting this task of modeling straight trajectories between two distributions.

3.2. Flow Matching

Flow Matching models are generative models that regress vector fields based on fixed conditional probability paths. Let $\mathbb{R}^d$ be the data space with data points x. Let $u_t(x) : [0,1] \times \mathbb{R}^d \to \mathbb{R}^d$ be the time-dependent vector field, which defines the ODE in the form of $dx = u_t(x)dt$, and let $\phi_t(x)$ denote the solution to this ODE with the initial condition $\phi_0(x) = x$.

The probability density path $p_t : [0,1] \times \mathbb{R}^d \to \mathbb{R}_{>0}$ depicts the probability distribution of x at timestep t with $\int p_t(x)dx = 1$. The pushforward function $p_t = [\phi_t]_\#(p_0)$ then transports the probability density path p along u from timestep 0 to t.

Assuming that $p_t(x)$ and $u_t(x)$ are known, and the vector field $u_t(x)$ generates $p_t(x)$, we can regress a vector field $v_\theta(t,x)$ parameterized by a neural network with learnable parameters θ using the Flow Matching objective

$$\mathcal{L}_{FM}(\theta) = \mathbb{E}_{t,p_t(x)}||v_\theta(t,x) - u_t(x)||. \tag{4}$$

While we generally do not have access to a closed form of u_t because this objective is intractable, Lipman et al. [31] showed that we can acquire the same gradients and therefore efficiently regress the neural network using the conditional Flow Matching (CFM) objective, where we can compute $u_t(x|z)$ by efficiently sampling $p_t(x|z)$,

$$\mathcal{L}_{CFM}(\theta) = \mathbb{E}_{t,q(z),p_t(x|z)}||v_\theta(t,x) - u_t(x|z)||, \tag{5}$$

with z as a conditioning variable and $q(z)$ the distribution of that variable. We parameterize v_θ as a U-Net [48], which takes the data sample x as input and z as conditioning information.

3.2.1 Gaussian Assumption

We first assume that the probability density path starts from p_0 with standard Gaussian distribution and ends up in a Gaussian distribution $\mathcal{N}(x_1, \sigma^2_{\min})$ that is smoothed around a data sample x_1 with minimal variance. In this case, the conditioning signal z would be x_1, and the optimal transportation path would be formulated as follows [31],

$$p_t(x|z) = \mathcal{N}(x|tx_1, (t\sigma_{\min} - t + 1)^2\mathbf{I}), \tag{6}$$

$$u_t(x|z) = \frac{x_1 - (1-\sigma_{\min})x}{1-(1-\sigma_{\min})t}, \tag{7}$$

$$\phi_t(x|z) = (1-(1-\sigma_{\min})t)x + tx_1. \tag{8}$$

The resulting CFM loss takes the form of

$$\begin{aligned}\mathcal{L}_{CFM}(\theta) &= \mathbb{E}_{t,z,p_t(x|z)}||v_\theta(t,\phi_t(x_0)) - \frac{d}{dt}\phi_t(x_0)|| \\ &= \mathbb{E}_{t,z,p(x_0)}||v_\theta(t,\phi_t(x_0)) - (x_1 - (1-\sigma_{\min})x_0)||.\end{aligned} \tag{9}$$

3.2.2 Data-Dependent Couplings

In our case, we also have access to the representation of a low-resolution image generated by a Diffusion model at inference time. It seems then intuitive to incorporate the inherent relationship between the conditioning signal and our target within the Flow Matching objective, as is also stated in [5]. Let x_1 denote a high-resolution image data sample. The conditioning signal $z := x_1$ remains unchanged from the previous formulation. Instead of randomly sampling from a Gaussian distribution in the naïve Flow Matching method, the starting point $x_0 = \mathcal{E}(x_1)$ corresponds to an encoded representation of the image, with $\mathcal{E}$ being a fixed encoder.

Like in the previously described case, we smooth around the data samples within a minimal variance to acquire the corresponding data distribution $\mathcal{N}(x_0, \sigma^2_{\min})$ and $\mathcal{N}(x_1, \sigma^2_{\min})$. The Gaussian flows can be defined by the equations

$$p_t(x|z) = \mathcal{N}(x|tx_1 + (1-t)x_0, \sigma^2_{\min}\mathbf{I}), \tag{10}$$

$$u_t(x|z) = x_1 - x_0, \tag{11}$$

$$\phi_t(x|z) = tx_1 + (1-t)x_0. \tag{12}$$

Notably, the optimal transport condition between the probability distributions $p_0(x|z)$ and $p_1(x|z)$ is inherently and automatically satisfied due to the low-resolution high-resolution data coupling. This automatically solves the dynamic optimal transport problem in the transition from low to high resolution within the Flow Matching paradigm, enabling more stable and faster training [59]. We name these

Flow Matching models with data-dependent couplings *Coupling Flow Matching* (CFM) models, and the CFM loss then takes the form of

$$\mathcal{L}_{CFM}(\theta) = \mathbb{E}_{t,z,p(x_0)}||v_\theta(t, \phi_t(x_0)) - (x_1 - x_0)||. \quad (13)$$

Note that the encoder $\mathcal{E}$ can be generalized to a wide range of image representations with spatial information, including but not limited to perceptual image compression [47], semantic segmentation maps [9] and more.

3.2.3 Noise Augmentation

Noise Augmentation is a technique for boosting generative models' performance introduced for cascaded Diffusion models [19]. The authors found that applying random Gaussian noise or Gaussian blur to the conditioning signal in super-resolution Diffusion models results in higher-quality results during inference. Drawing inspiration from this, we also implement Gaussian noise augmentation on x_0. Following the notation from variance-preserving DMs, we noise x_0 according to the cosine schedule first proposed in [42]. In line with [19], we empirically discover in our experiments that incorporating a specific amount of Gaussian noise enhances performance. We hypothesize that including a small amount of Gaussian noise smoothes the base probability density p_0 so that it remains well-defined over the higher-dimensional space. Note that this noise augmentation is only applied to x_0 but not to the conditioning information z, since the model relies on the precise conditioning information to construct the straight path.

3.2.4 Latent Flow Matching

In order to reduce the computational demands associated with training Flow Matching models for high-resolution image synthesis, we take inspiration from [12, 47] and utilize an autoencoder model that provides a compressed latent space that aligns perceptually with the image pixel space similar to LDMs. By training in the latent space, we get a three-fold advantage: i) The computational cost associated with the training of Flow Matching models is reduced substantially, thereby enhancing the overall efficiency of the training process. ii) Leveraging the latent space unlocks the potential to synthesize images with significantly increased resolution efficiently. iii) The lower resolution in the latent space also yields quicker inference speed in the synthesis task.

3.3. High-Resolution Image Synthesis

Overall, our approach integrates all the components discussed above into a cohesive synthesis pipeline, as depicted in detail in Fig. 3. We start from a DM for content synthesis and move the generation to a latent space with a pretrained VAE encoder, which optimizes memory usage and enhances inference speed. To further alleviate the computational load of DM and achieve additional acceleration, we adopt a relatively compact DM that produces compressed information. Subsequently, the FM model projects the compressed information to a high-resolution latent image with a straight conditional probability path. Finally, we decompress the latent space using a pretrained VAE decoder. Note that the VAE decoder performs well across various resolutions, as we demonstrate in Fig. 16 and Tab. 6 in the supplementary materials.

The integration of FM with DMs in the latent space presents a promising approach to address the trade-off between flexibility and efficiency in modeling the dynamic image synthesis process. The inherent stochasticity within a DM's sampling process allows for a more nuanced representation of complex phenomena, while the FM model exhibits greater computational efficiency, which is useful when handling high-resolution images, but lower flexibility and image fidelity as of yet when it comes to image synthesis [31]. By combining them in the pipeline, we benefit from the flexibility of the DM while capitalizing on the efficiency of FM as well as VAE.

4. Experiments

Combining a diffusion synthesis model and a flow matching super-resolution model yields great synthesis capability in the high-resolution domain, as we show qualitatively and quantitatively in this section. We first demonstrate the effectiveness of the combination, and then provide an extensive ablation and analysis of our CFM model. Subsequently, we assess our model in comparison to the current state-of-the-art diffusion approach.

4.1. Datasets

The general dataset that we use for training and ablations is FacesHQ, a compilation of CelebA-HQ [23] and FFHQ [24], as employed in previous work such as [13]. This dataset serves as a foundation for evaluating various aspects of our model. We also include ablations on LHQ [54], which contains 90k high-resolution landscape images.

For the general flow matching (FM) model, particularly targeted towards combination with latent diffusion models, we utilize the Unsplash dataset [3], which provides diverse and high-quality images for training our model. We present inference results of our general FM model on a high-resolution subset of LAION 5B [52].

4.2. Boosting LDM with FM

Aggregating the power of LDM and FM results in remarkable performance in image synthesis. Combining the flexibility of LDM with the intrinsic characteristics of FM

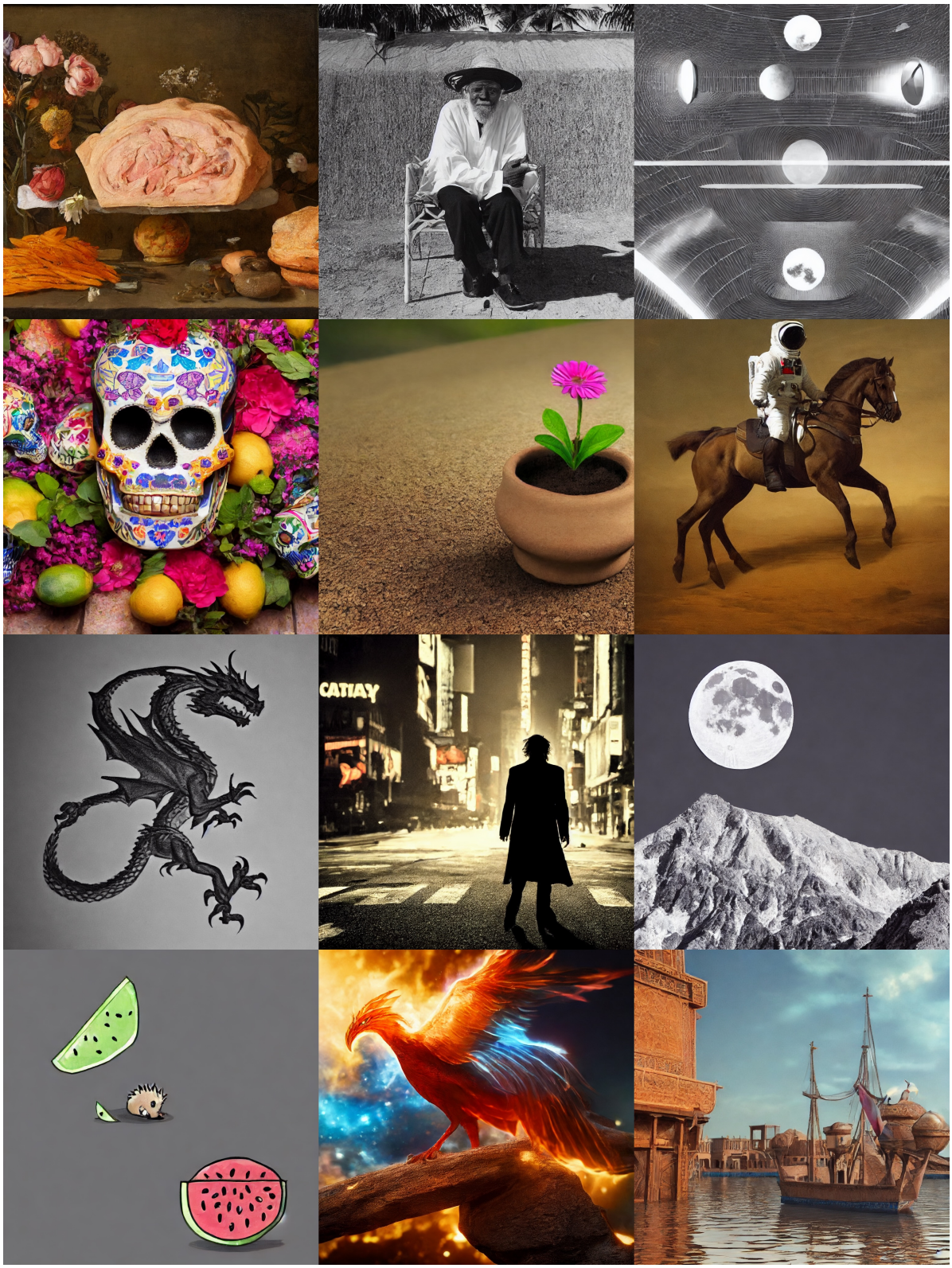

Figure 4. Uncurated samples from the Coupling Flow Matching model on top of Stable Diffusion V1.5 [47] using a classifier-free guidance scale of 7.5. Samples are generated in latent space 64^2 and up-sampled with CFM from 64^2 to 128^2. The resulting images have a resolution of 1024×1024.

achieves an optimal trade-off between computational efficiency and visual fidelity.

Since LDMs are typically trained and evaluated on fixed sizes, we follow [21] and apply attention scaling on SD to synthesize images of varying resolutions. We also finetune the small LDM model which outputs images at resolutions of 256^2 px to further ensure image fidelity. More details can be found in Appendix Sec. 6.2. We plot the inference speed for 1024^2 px image in Fig. 1, where we visualize the time taken by LDM and FM distinctively. Note that the LDM's inference time scales quadratically with increasing resolution, and the inference is almost impractical for a real-time inference for a latent space of 128^2. Conversely, the CFM model shows remarkably fast inference speed, which emphasises the efficiency improvements achieved by its integration for high-resolution image synthesis.

We demonstrate the models' synthesized image quality in Fig. 5. Both combinations of LDM and FM surpass the performance of the attention-scaled LDM-128 baseline in terms of inference time and image quality. The intermediate resolution of 32^2 yields faster inference, while the intermediate resolution of 64^2 maintains better image fidelity. Note that we achieve near-optimal results with as few as 10 ODE steps for the CFM model, which adds minimal computational overhead to the underlying DM. Note that we selectively use a subset of 1k samples from LAION to highlight variations in NFE for a fair comparison across all models. This choice is for rapid evaluation purposes, and the FID is expected to decrease with a larger sample size. We present a selection of image samples of LDM-64 and CFM $64^2 \rightarrow 128^2$ in Fig. 4. Please refer to Fig. 11 and Fig. 14 in the Appendix for more samples.

We also investigate the results for consistency models [58], which significantly accelerate the inference process by distilling information from a diffusion model. The LCM-LoRA model [38] finetuned on SD V1.5 [47] produces 512×512 resolution images. We can further improve this model to 1k resolution images with our CFM model. We present our synthesised results in Fig. 2. The inference time for a batch of four samples is 1.388 seconds. The LCM-LoRA SDXL model [43] fails to produce images with similar fidelity at the same resolution in the same time range.

4.3. FM Model Ablation

Upsampling Methods As our flow matching model requires that x_0 and x_1 share the same dimensionality, we need to upsample the latent code. In this context, we conduct an ablation study comparing two upsampling methods: nearest neighbor and bi-linear upsampling. The results are presented in Tab. 1. While the choice between these methods does not appear to be decisive based on SSIM and PSNR metrics, we observe generally lower FID [16] values with nearest neighbor upsampling.

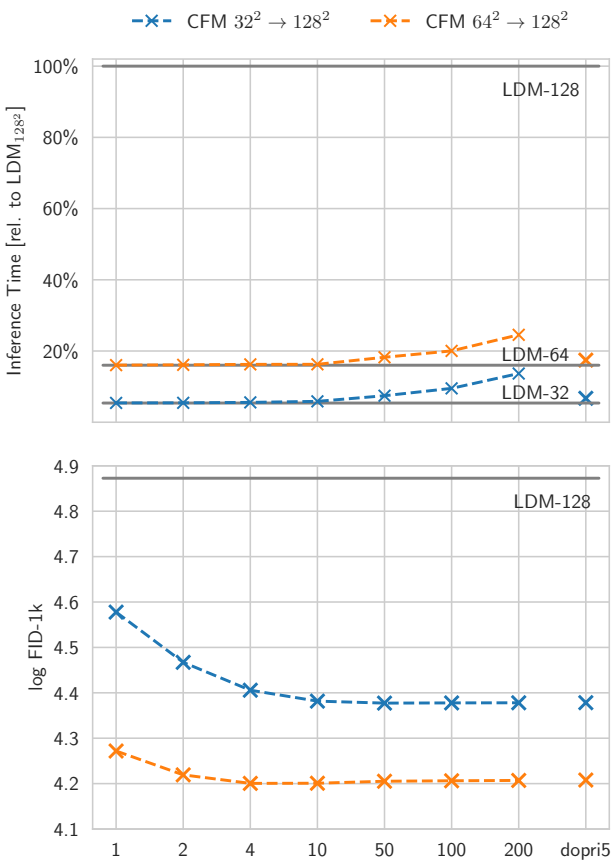

Figure 5. Inference time per model part (LDM or CFM) and FID for synthesizing images with resolution 1024×1024 for different number of function evaluations (NFE). The upper plot refers to the average inference time relative to LDM-128, while the lower plot indicates log FID for 1k samples. *dopri5* corresponds to the Dormand-Prince adaptive step-size solver. For LDM-128 we do not use any CFM model.

Flow Matching and Coupling Flow Matching In this ablation study, we systematically explore two variations of Flow Matching models – one incorporating the Gaussian assumption (FM) and the other incorporating data-dependent coupling (CFM). We evaluated these two variants quantitatively (Tab. 1) and qualitatively (Fig. 7), where we observed that the CFM models with data-dependent coupling readily outperform the ones without. We provide more information about the noise augmentation process in Tab. 3. Notably, in the specific upsampling scenario from 256^2 to 1024^2, we observe an optimal configuration with a noising timestep of 200. The introduction of Gaussian noise proves beneficial as it imparts a smoothing effect on the input probability path, resulting in improved performance. However, excessive Gaussian noise can lead to the loss of valuable information, subsequently deteriorating the data-dependent coupling and reverting the model's behavior to the Gaussian

Model	CelebA-HQ			FFHQ			FacesHQ		
	SSIM ↑	**PSNR** ↑	**FID** ↓	**SSIM** ↑	**PSNR** ↑	**FID** ↓	**SSIM** ↑	**PSNR** ↑	**FID** ↓
CFM-nearest	0.75	26.47	4.00	0.70	25.47	3.65	0.72	25.80	2.75
CFM-bilinear	0.75	26.43	4.58	0.70	25.50	3.94	0.70	25.49	3.93
FM-nearest	0.75	26.47	7.91	0.70	25.47	8.16	0.71	25.80	6.84
FM-bilinear	0.75	26.47	8.10	0.69	25.47	7.98	0.71	25.80	6.78

Table 1. Ablation of up-sampling methods for Flow Matching in the latent space. While the choice between these methods does not appear to be decisive based on SSIM and PSNR metrics, we observe generally lower FID values with nearest neighbor upsampling which also matches the visual inspection of generated results shown in the supplementary.

Figure 6. Results of different latent space up-sampling strategies from 32×32 to 128×128. *LR* corresponds to bilinear upsampling of the low-resolution image, *Reg* refers to the regression baseline. *FM* and *CFM* correspond to Flow Matching and Coupling Flow Matching with a noising step of 400, respectively.

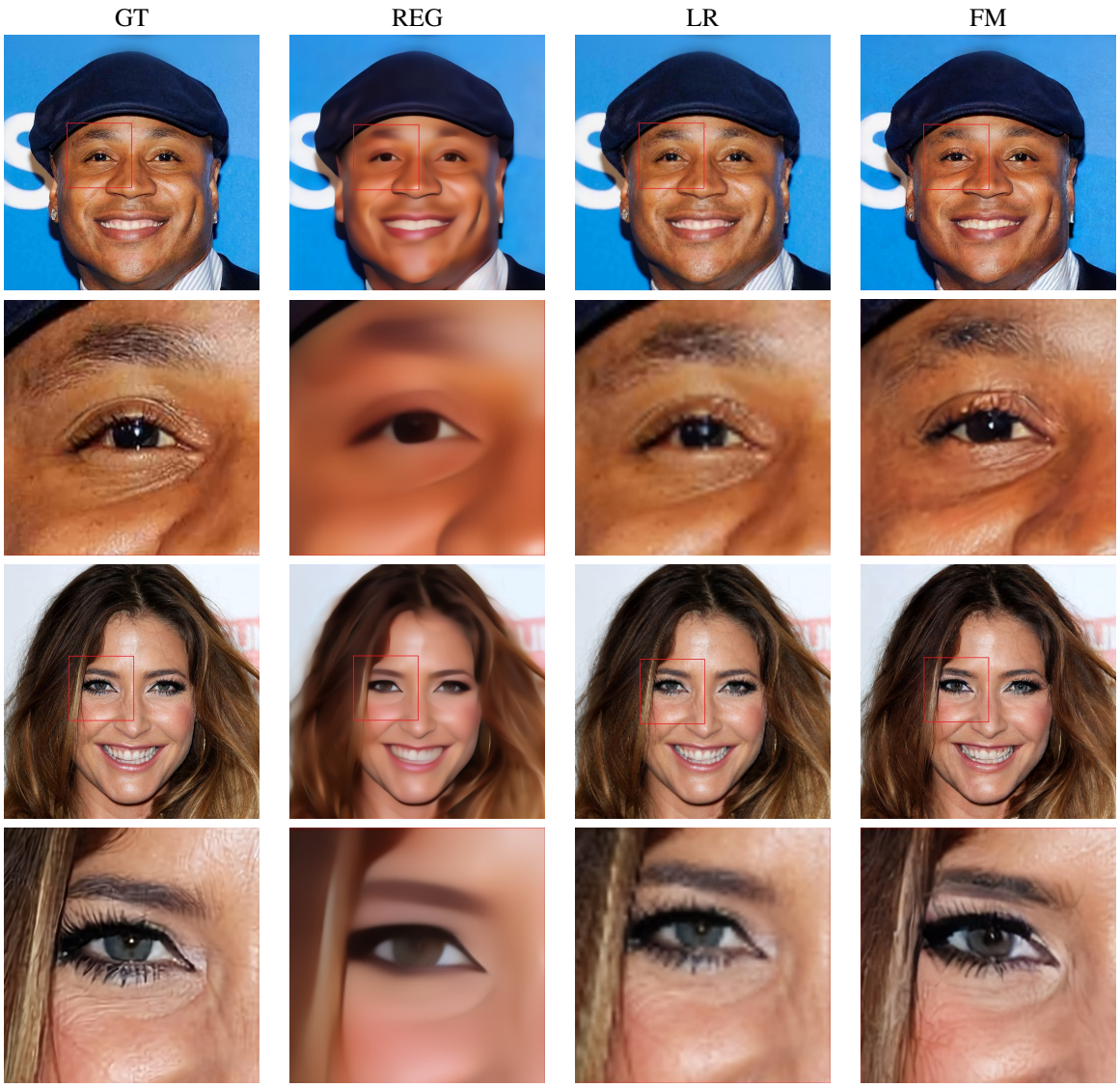

Figure 7. Qualitative comparison of the Flow Matching (FM) model to the upsampled low resolution image (LR) the regression model (REG) and the ground truth (GT) using zoomed-in patches. The FM patches are significantly sharper and display more fine-grained structure and details.

Model	CelebA-HQ			FFHQ			FacesHQ		
	SSIM ↑	**PSNR** ↑	**FID** ↓	**SSIM** ↑	**PSNR** ↑	**FID** ↓	**SSIM** ↑	**PSNR** ↑	**FID** ↓
Reg	0.81	26.99	56.15	0.77	26.29	48.22	0.78	26.52	48.32
FM	0.75	26.47	7.91	0.70	25.47	8.16	0.71	25.80	6.84
CFM	0.75	26.47	4.00	0.70	25.47	3.65	0.72	25.80	2.75

Table 2. Metric results for regression (*Reg*), Flow Matching (*FM*), and Coupling Flow Matching (*CFM*) corresponding to Fig. 6.

Figure 8. Intermediate results along the ODE trajectory with Euler ODE solver and 100 function evaluations. From left to right, the number of function evaluations is 0 (upsampled latent w/o FM), 25, 50, 75, and 100.

assumption. This finding underscores the delicate balance required in incorporating noise for optimal model performance.

Intermediate Results along the ODE Trajectory Fig. 8 shows intermediate results along the ODE trajectory. It can be seen that the CFM model gradually transforms the image representation from one with grid artefacts due to nearest neighbour upsampling to its high-resolution image counterpart.

4.4. Comparison to Baselines

We compare our Coupling Flow Matching method to the diffusion-based up-sampling method SR3 [50], as well as a simple regression baseline, trained with a simplep $L2$ loss, on the FacesHQ dataset. For a fair comparison, we keep the architecture and hyperparameters fixed for all three models. We obtain the latent codes by using the pre-trained KL-regularized autoencoder from [47] and perform nearest neighbor up-sampling to fit the dimensionality of our starting condition x_0.

Upsampling Results Fig. 6 shows results of different latent space up-sampling strategies from 32^2 to 128^2, increasing the final output resolution from 256^2 to 1024^2. We can observe that the simple regression baseline yields blurry results, which is also reflected in the metrics (see Tab. 2). In contrast, both flow matching approaches – Coupling Flow Matching (CFM) and Flow Matching starting from a noise distribution (FM) – yield better results. Also, metric-wise, both outperform the simple regression baseline by a large margin. When directly comparing the low-resolution image at size 256^2 with the decoded output of our Flow Matching model (Fig. 6), we can observe that FM adds fine-grained details to the image not visible in the lower-resolution counterpart.

Training Efficiency Based on optimal transport theory, the training of a constant velocity field presents a more straightforward training objective when contrasted with the intricate high-curvature probability paths found in diffusion models [12, 31]. This distinction often translates to slower training convergence and potentially sub-optimal trajectories for DMs, which could detrimentally impact both, training duration and overall model performance. Fig. 9 shows the FID over the course of training for the diffusion-based SR3 and our Flow Matching model. We can clearly observe that the Flow Matching model achieves a lower FID more rapidly and produces superior results, compared to the SR3 model. Fig. 10 further demonstrates that after 100k iterations and for different number of function evaluations (NFE), we consistently attain a lower FID compared to the

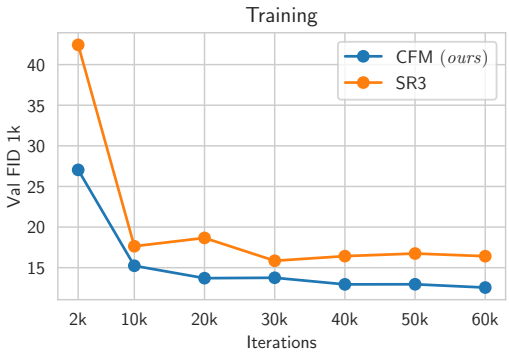

Figure 9. Comparison of diffusion-based SR3 [50] and our Coupling Flow Matching (CFM) over the training for $4\times$ up-sampling of the latent codes from $32^2 \rightarrow 128^2$. Decoded output resolution is 1024^2. Architecture and hyperparameters are kept fixed for both. FID evaluated on 1k samples from the validation set. We use DDIM sampling with 50 steps for the diffusion-based model and the Euler ODE solver with 50 steps for CFM.

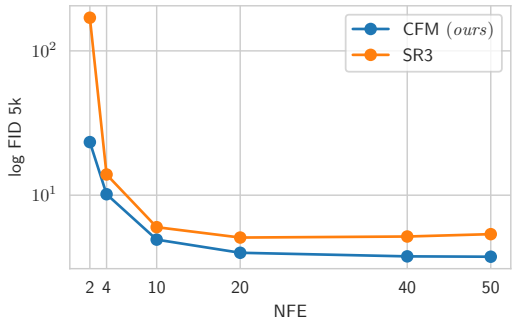

Figure 10. Comparison of diffusion-based SR3 [50] and our Coupling Flow Matching (CFM) for different number of function evaluations (NFE) for $4\times$ up-sampling of the latent codes from $32^2 \rightarrow 128^2$. Decoded output resolution is 1024^2. Architecture and hyperparameters are kept fixed for both. FID evaluated on 5k samples from the validation set. We use DDIM [56] for diffusion-based sampling and the Euler ODE solver for the CFM model.

Model	CelebA-HQ			FFHQ			FacesHQ		
	SSIM ↑	**PSNR** ↑	**FID** ↓	**SSIM** ↑	**PSNR** ↑	**FID** ↓	**SSIM** ↑	**PSNR** ↑	**FID** ↓
CFM-100	0.71	25.30	8.27	0.73	26.14	5.84	0.75	26.41	5.47
CFM-200	0.75	26.39	3.90	0.70	25.48	3.14	0.72	25.78	2.38
CFM-400	0.75	26.47	4.00	0.70	25.47	3.65	0.72	25.80	2.75
CFM-600	0.75	26.51	5.18	0.70	25.52	5.14	0.72	25.85	4.04
CFM-999	0.74	26.36	7.72	0.69	25.35	7.53	0.71	25.69	6.38

Table 3. Quantative evaluation on different noising steps for coupling flow matching (CFM) for $4\times$ upsampling from 256^2 to 1024^2 px. We evaluate the FID on the full dataset (5k, 10k, and 15k samples respectively).

diffusion-based SR3 model. Taken together, these findings underscore the training efficiency of Flow Matching over Diffusion models and its superior performance for the upsampling task.

5. Conclusion

Our work introduces a novel and effective approach to high-resolution image synthesis, combining the generation diversity of Diffusion Models, the efficiency of Flow Matching, and the effectiveness of convolutional decoders. Strategically integrating Flow Matching models between a standard latent Diffusion model and the convolutional decoder enables a significant reduction in the computational cost of the generation process by letting the expensive Diffusion model operate at a lower resolution and up-scaling its outputs using an efficient Flow Matching model. Our Flow Matching model efficiently enhances the resolution of the latent space without compromising quality. Our approach complements DMs with their advancements and is orthogonal to their recent enhancements such as sampling acceleration and distillation techniques e.g., LCM [38]. This allows for mutual benefits between different approaches and ensures the smooth integration of our method into existing frameworks.

References

[1] LAION-Aesthetics | https://laion.ai/blog/laion-aesthetics. 1

[2] OpenAI | https://github.com/openai/guided-diffusion/blob/main/guided_diffusion/unet.py. 2

[3] Unsplash | https://unsplash.com/data. 6

[4] Michael S Albergo and Eric Vanden-Eijnden. Building normalizing flows with stochastic interpolants. In *ICLR*, 2023. 3, 5

[5] Michael S Albergo, Mark Goldstein, Nicholas M Boffi, Rajesh Ranganath, and Eric Vanden-Eijnden. Stochastic interpolants with data-dependent couplings. *arXiv preprint arXiv:2310.03725*, 2023. 3, 5

[6] Sepehr Sameni Aram Davtyan and Paolo Favaro. Efficient video prediction via sparsely conditioned flow matching. In *ICCV*, 2023. 3

[7] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, et al. ediffi: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022. 2

[8] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *CVPR*, 2023. 3, 4

[9] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 6

[10] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, et al. Fast training of diffusion transformer for photorealistic text-to-image synthesis. *arXiv preprint arXiv:2310.00426*, 2023. 1

[11] Ricky T. Q. Chen. torchdiffeq, 2018. 1

[12] Quan Dao, Hao Phung, Binh Nguyen, and Anh Tran. Flow matching in latent space. *arXiv preprint arXiv:2307.08698*, 2023. 3, 6, 11

[13] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021. 6

[14] Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7346–7356, 2023. 3

[15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 3

[16] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 8

[17] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 4

[18] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 3, 4

[19] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *The Journal of Machine Learning Research*, 23(1):2249–2281, 2022. 1, 6, 2

[20] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. In *arXiv*, 2022. 3, 4

[21] Zhiyu Jin, Xuli Shen, Bin Li, and Xiangyang Xue. Training-free Diffusion Model Adaptation for Variable-Sized Text-to-Image Synthesis, 2023. arXiv:2306.08645 [cs, eess]. 1, 8

[22] Minguk Kang, Jun-Yan Zhu, Richard Zhang, Jaesik Park, Eli Shechtman, Sylvain Paris, and Taesung Park. Scaling up gans for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10124–10134, 2023. 3

[23] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017. 6

[24] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2019. 3, 6

[25] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *NeurIPS*, 2022. 3

[26] Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. In *NeurIPS*, 2021. 3

[27] Matthew Le, Apoorv Vyas, Bowen Shi, Brian Karrer, Leda Sari, Rashel Moritz, Mary Williamson, Vimal Manohar, Yossi Adi, Jay Mahadeokar, et al. Voicebox: Text-guided multilingual universal speech generation at scale. In *arXiv*, 2023. 3

[28] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017. 3

[29] Haoying Li, Yifan Yang, Meng Chang, Shiqi Chen, Huajun Feng, Zhihai Xu, Qi Li, and Yueting Chen. Srdiff: Single image super-resolution with diffusion probabilistic models. *Neurocomputing*, 479:47–59, 2022. 3

[30] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22511–22521, 2023. 3

[31] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *ICLR*, 2023. 2, 3, 5, 6, 11

[32] Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo Mandic, Wenwu Wang, and Mark D Plumbley. Audioldm: Text-to-audio generation with latent diffusion models. In *ICML*, 2023. 3

[33] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds. In *ICLR*, 2022. 3

[34] Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *ICLR*, 2023. 3

[35] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *NeurIPS*, 2022. 3

[36] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver++: Fast solver for guided sampling of diffusion probabilistic models. *arXiv preprint arXiv:2211.01095*, 2022. 3

[37] Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference. *arXiv preprint arXiv:2310.04378*, 2023. 3

[38] Simian Luo, Yiqin Tan, Suraj Patil, Daniel Gu, Patrick von Platen, Apolinário Passos, Longbo Huang, Jian Li, and Hang Zhao. Lcm-lora: A universal stable-diffusion acceleration module. *arXiv preprint arXiv:2311.05556*, 2023. 2, 3, 8, 12

[39] Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14297–14306, 2023. 3

[40] Kirill Neklyudov, Rob Brekelmans, Daniel Severo, and Alireza Makhzani. Action matching: Learning stochastic dynamics from samples. In *ICML*, 2023. 3

[41] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 3

[42] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *ICML*, 2021. 3, 4, 6, 2

[43] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 1, 2, 3, 8

[44] Konpat Preechakul, Nattanat Chatthee, Suttisak Wizadwongsa, and Supasorn Suwajanakorn. Diffusion autoencoders: Toward a meaningful and decodable representation. In *CVPR*, 2022. 3

[45] Markus N Rabe and Charles Staats. Self-attention does not need $o(n^2)$ memory. *arXiv preprint arXiv:2112.05682*, 2021. 1

[46] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1 (2):3, 2022. 1, 3

[47] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 1, 2, 3, 4, 6, 7, 8, 11

[48] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015. 5

[49] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022. 1, 3, 4

[50] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *TPAMI*, 45(4):4713–4726, 2022. 3, 11, 4

[51] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *ICLR*, 2022. 3

[52] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022. 6

[53] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022. 4

[54] Ivan Skorokhodov, Grigorii Sotnikov, and Mohamed Elhoseiny. Aligning latent and image spaces to connect the unconnectable. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14144–14153, 2021. 6

[55] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, 2015. 3

[56] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021. 1, 3, 11

[57] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *ICLR*, 2021. 3

[58] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *ICML*, 2023. 3, 8

[59] Alexander Tong, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Kilian Fatras, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. In *ICML*, 2023. 3, 5, 2

[60] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European conference on computer vision (ECCV) workshops*, pages 0–0, 2018. 3

[61] Zhisheng Xiao, Karsten Kreis, and Arash Vahdat. Tackling the generative learning trilemma with denoising diffusion gans. *arXiv preprint arXiv:2112.07804*, 2021. 1

[62] Zeyue Xue, Guanglu Song, Qiushan Guo, Boxiao Liu, Zhuofan Zong, Yu Liu, and Ping Luo. Raphael: Text-to-image generation via large mixture of diffusion paths. *arXiv preprint arXiv:2305.18295*, 2023. 3

[63] Zongsheng Yue, Jianyi Wang, and Chen Change Loy. Resshift: Efficient diffusion model for image super-resolution by residual shifting. *arXiv preprint arXiv:2307.12348*, 2023. 3

[64] Kai Zhang, Jingyun Liang, Luc Van Gool, and Radu Timofte. Designing a practical degradation model for deep blind image super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4791–4800, 2021. 3

[65] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023. 3

Boosting Latent Diffusion with Flow Matching

Supplementary Material

We initiate our analysis by ablations on the flow matching model, systematically exploring its individual components and assessing their impact on the overall performance. Subsequently, we delve into a comprehensive examination of the combined effects of the flow matching and diffusion models. This examination considers both image quality and the efficiency gains achieved through this integration.

6. Quantitative Experimental Results

6.1. Discussion on Flow Matching

Effects of ODE Solver Steps on Image Quality We use the *torchdiffeq* [11] library for ordinary differential equation (ODE) solvers implemented in PyTorch. Tab. 4 demonstrates the ability of our flow matching model to produce strong results based on FID with as few as 10 function evaluations with additional speedups when compared against diffusion models. This highlights the efficiency and advantage of Flow Matching models in contrast to score-based models, which typically require more function evaluations due to its inherent stochasticity. Fig. 12 shows the average time needed for up-sampling a latent code from 32×32 to 128×128 on a single NVIDIA A100 GPU with batch-size of 4 and Coupling Flow Matching. At 10 steps, the FID score plateaus, emphasizing the minimal gains from additional steps and empirically demonstrating that modeling direct trajectories accelerates sampling speed.

	FacesHQ-1k			
NFE	**SSIM** ↑	**PSNR** ↑	**FID** ↓	**Time** (s) ↓
1	0.81	27.41	66.24	395
2	0.81	27.86	33.89	399
4	0.79	27.71	17.69	413
10	0.78	27.10	11.53	465
50	0.75	26.50	10.31	718
100	0.75	26.50	10.30	1,046
1000	0.75	26.32	10.28	7,061
dopri5	0.75	26.31	10.34	996

Table 4. We ablate the CFM-200 model and compare different number of function evaluations (NFE) during inference for the *Euler* method, as well as the adaptive step-size Dormand-Prince solver. We evaluate the results on a subset of 1k samples of FacesHQ and present the total inference time on a single NVIDIA A100 GPU.

We additionally illustrate the outcomes of image up-sampling for various number of function evaluations (NFE) in Fig. 13. An intriguing observation emerges when working with an exceptionally low NFE, which leads to higher PSNR and SSIM but with a notably increased blurriness. These results highly resemble the regression baseline model. With increasing number of steps images get sharper, with marginal to no perceptual differences between 10 and 50 NFEs. This again emphasizes the low number of function evaluations required to obtain reasonable high-resolution results from low-resolution conditioning information.

6.2. DMs for lower Resolutions

To evaluate the performance of our approach, we perform flow matching on also low-dimensional latents. The standard SD V1.5 [47] was trained to work on latents with a dimensionality of 64^2 pixels. After decoding the latent, this results in generated images with a resolution of 512^2 pixels. Sampling latents of a different dimensionality is also possible but results in degraded performance, which is particularly pronounced for sampling lower-resolution latents. This deterioration can be partially mitigated by changing the scale of the self-attention layers from the standard $\sqrt{1/d}$ to $\sqrt{\log_T N/d}$ where d is the inner attention dimensionality and T and N the number tokens during the training and inference phase respectively [21]. We use this to rescale the attention layers of SD V1.5 for 32^2 latents. Additionally, we finetune it on images of LAION-Aesthetics V2 6+ [1] rescaled to 256^2 pixels for one epoch, a learning rate of $1e{-}5$, and batch size of 256.

We provide fine-tuned LDM metrics on the LAION dataset in Tab. 5. We found that FID pleataus with a classifier-free guidance scale at 7.0. Consequently, we adopt this value for small resolution (32^2 in the latent space), text-guided image synthesis. We illustrate the uncurated samples for the upsampled images in Fig. 14.

7. Qualitative Experimental Results

Fig. 18 shows additional, zoomed in results of our FacesHQ Coupling Flow Matching model, where we up-sample the latent code of 256^2 images from 32^2 to 128^2 with the Dormand-Prince ODE solver. In Fig. 19 and Fig. 20 we visualize more, non-curated results from our Coupling Flow Matching model on the FacesHQ dataset and the LHQ dataset respectively.

Fig. 17 show generated text-to-image samples in 2048^2 px using the LDM together with CFM up-sampling in latent space with an intermediate resolution of 512^2, produced by SD V1.5.

Figure 11. Samples from the Coupling Flow Matching model on top of Stable Diffusion V1.5 [47] using a classifier-free guidance scale of 7.5. Samples are generated in latent space 64^2 and up-sampled with CFM from 64^2 to 128^2. The resulting image has a resolution of 1024×1024. Best viewed when zoomed-in.

8. Architecture and Implementation Details

In our implementation of FM, we follow Tong et al. [59] and set the minimal variance either as $\sigma_{\text{min}} = 1e{-}8$ or $\sigma_{\text{min}} = 1e{-}4$. We employ the cosine noising schedule [42] for noise augmentation [19], and $t \in [1, 999]$ resembles the corresponding timestep. We adopt the widely used U-Net architecture as implemented in the OpenAI GitHub repository [2]. Implementation details and parameterization for the UNet used for our Coupling Flow Matching models can be found in Tab. 8.

KL Autoencoder We employ the KL-regularized autoencoder from Stable Diffusion V1.5 [47] to compress images to their respective latent representations. We also shed light on how well the pre-trained autoencoder from Stable Diffusion perform on smaller and larger resolution. We focus here on the autoencoder architecture that we employ for latent compression, since it is crucial to ensure that the latent space captures perceptual information at various resolution. Thanks to the convolutional nature, the autoencoder exhibits the capability to encode images at multiple scales. To systematically evaluate its effectiveness, we present re-

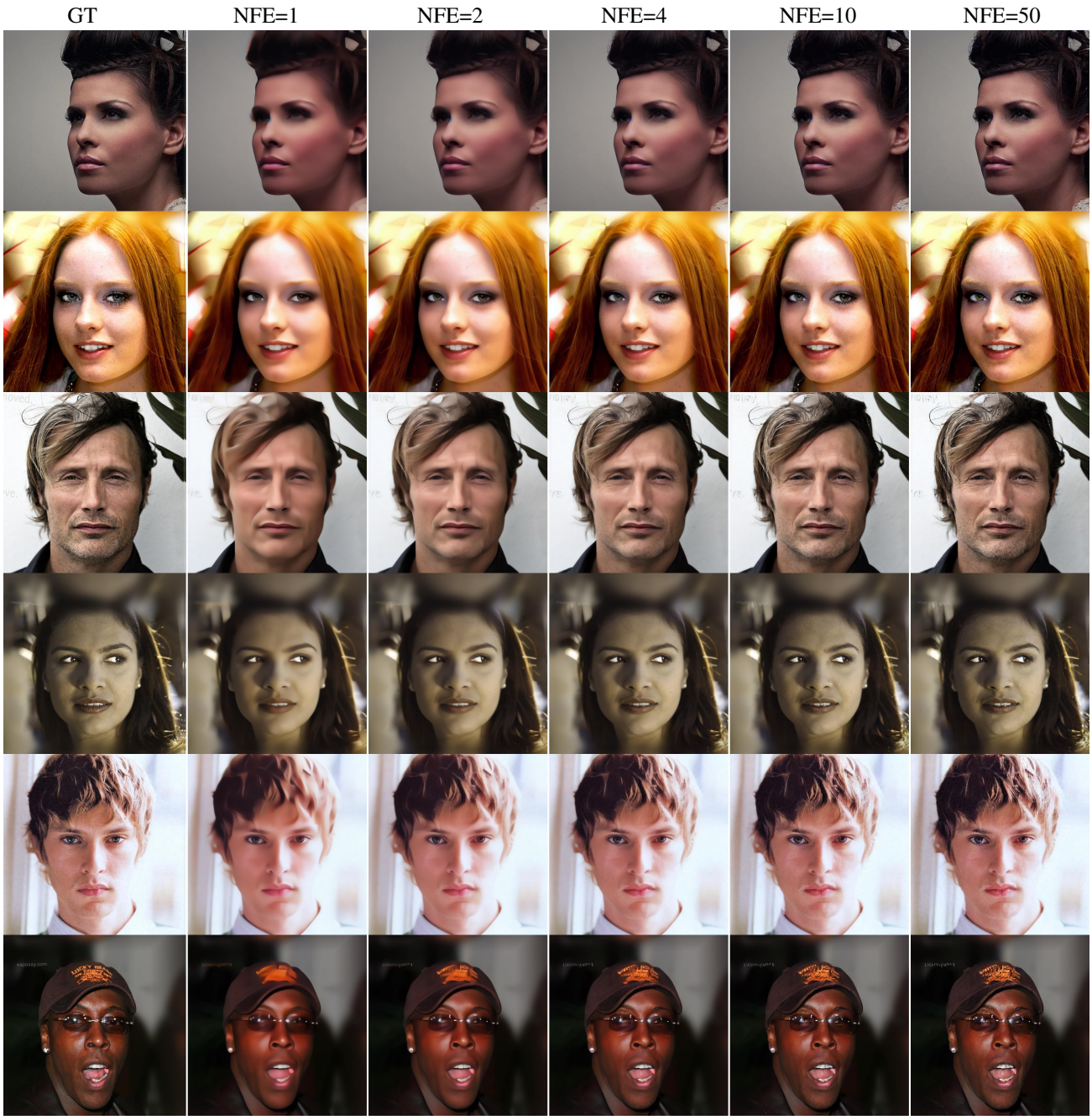

Figure 13. Sample quality of different ODE solvers and different number of function evaluations (NFE). From left to right, 1st column represents the ground truth, high-resolution images. From 2nd column on, we show the results for NFE $= 1, 2, 4, 10, 50$ with the Euler fixed step-size ODE solver.

sults using metrics such as FID, PSNR, and SSIM for images of different sizes. The detailed metrics are presented in Tab. 6, and the FID is calculated with a representative sample size of $10,000$ images. Fig. 16 additionally shows visual results. Our analysis underscores the autoencoder's proficiency across multiple resolutions. Despite potential shortcomings in low-resolution compression, we contend that the FM model can adeptly learn to transition from this distribution to the latent space of the high-resolution image, thereby mitigating deficiencies in low-resolution performance.

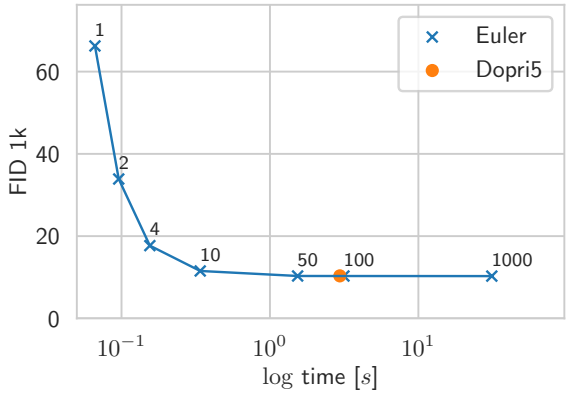

Figure 12. Average computation time on a single NVIDIA A100 GPU with batch-size 4 for different number of function evaluations with the Euler method and the Dormand-Prince adaptive step-size ODE solver on the FacesHQ dataset.

CFG	FID ↓	CLIP ↑
1	41.95	0.157
3	17.13	0.214
5	14.03	0.231
7	13.47	0.239
9	13.50	0.243

Table 5. Results of our fine-tuned Latent Diffusion Model on the LAION dataset for different Classifier-free guidance scales. [17]

Image Size	LAION-10k SSIM ↑	PSNR ↑	FID ↓
256	0.82	26.42	2.47
512	0.86	28.65	1.28
1024	0.88	30.72	0.84
2048	0.88	32.20	0.60

Table 6. Evaluation of the pre-trained autoencoder performance under different image sizes.

Model	Big (306M) SSIM ↑	PSNR ↑	FID ↓	Small (113M) SSIM ↑	PSNR ↑	FID ↓
FM	0.59	22.90	11.22	0.59	22.84	10.90
CFM-400	0.59	22.72	6.30	0.59	22.87	6.38
CFM-200	0.60	22.98	4.92	0.60	23.02	5.20

Table 7. LHQ ablation for $256^2 \rightarrow 1024^2$ super-resolution results.

	FacesHQ	Unsplash	LHQ
Autoencoder f	8	8	8
z-shape	$4 \times 64 \times 64$	$4 \times 64 \times 64$	$4 \times 64 \times 64$
Model size	113M	306M	306M
Channels	128	128	128
Depth	3	3	3
Channel multiplier	1, 2, 3, 4	1, 2, 4, 8	1, 2, 4, 8
Attention resolutions	16	16	16
Head channels	64	64	64
Number of heads	4	4	4
Optimizer	Adam	Adam	Adam
Batch size	96	768	128
Learning rate	1e-4	1e-4	1e-4

Table 8. Hyperparameters and number of parameters for our FMs and DMs. We use the exact same architecture for our FM and the SR3 [50] upsampling model to ensure a fair comparison. We trained the SR3 model with 1000 diffusion steps and a cosine noise schedule [42].

Figure 14. Uncurated samples from the Coupling Flow Matching model on top of finetuned Stable Diffusion for lower resolution using a classifier-free guidance scale of 7.0. Samples are generated in latent space 32^2 and up-sampled with CFM from 32^2 to 128^2. The resulting images have a resolution of 1024×1024.

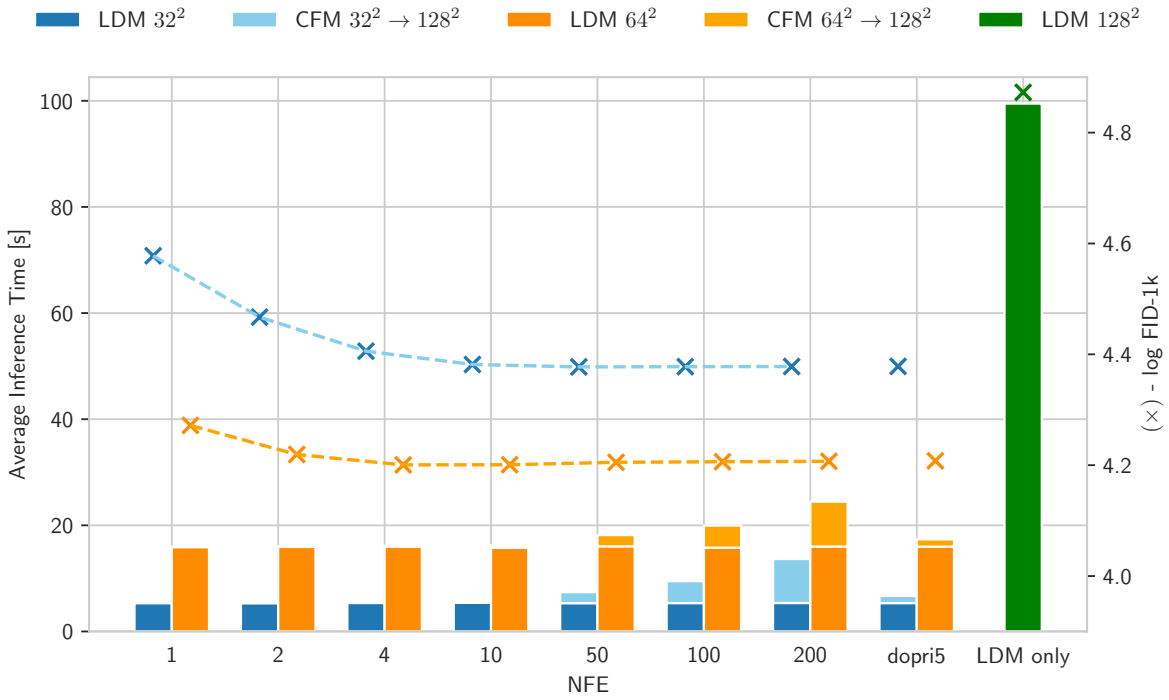

Figure 15. Absolute inference time per model part (LDM or CFM) and FID for synthesising images with resolution 1024×1024 for different number of function evaluations (NFE). Left axis refers to average inference time in seconds, right axis to log FID for 1k samples. *Dopri5* corresponds to the Dormand-Prince adaptive step-size solver. For LDM 128^2 we do not use any CFM model.

Figure 16. Comparison between reconstructed images (middle 256^2 and right column 2048^2) and their high-resolution original input (left column) using the pre-trained autoencoder. We can observe that the (i) pre-trained autoencoder is able to encode and decode images at different scales. (ii) It cannot reconstruct faces correctly in low resolution. (iii) The artifacts diminish with a higher resolution. Best viewed when zoomed in.

Figure 17. Samples from the Coupling Flow Matching model on top of LCM-LoRA in 2k resolution. Samples are generated in latent space 64^2 and up-sampled with CFM from 64^2 to 256^2. The resulting image has a resolution of 2048×2048. Best viewed when zoomed-in.

Figure 18. Qualitative comparison of the Flow Matching (FM) model to the upsampled low resolution image (LR) the regression model (REG) and the ground truth (GT) using zoomed-in patches. The FM patches are significantly sharper and display more fine-grained details.

Figure 19. Uncurated samples from the Coupling Flow Matching model on FacesHQ. Up-sampling in latent space from 32×32 to 128×128. Resulting image with resolution of 1024×1024 (bottom) compared to the low-resolution image, up-sampled to 1024×1024 (top). Best viewed when zoomed-in.

Figure 20. LHQ super-resolution samples. Left: low-resolution ground truth image bi-linearly up-sampled. Right: synthesized high resolution image. Best viewed when zoomed in.