A HIGH-FIDELITY TEXT-TO-IMAGE EDITING APPROACH: GAUSSIAN FLUID DYNAMICS IN LATENT SPACE

Anonymous authors
Paper under double-blind review

ABSTRACT

Text-guided image editing with generative models has recently achieved remarkable progress, yet the underlying dynamics of latent space manipulations remain insufficiently explored. In this work, we propose a perspective that models the latent space of generative models as a high-dimensional Gaussian fluid. Specifically, each latent dimension is regarded as a directional axis of the fluid, and the movement of data points within this space is governed by three interacting forces: a driving force that enforces semantic editing objectives, a resistance force that preserves data consistency, and a central constraint that maintains generation quality. We formalize this process through the Navier–Stokes equations, enabling a principled formulation of latent space dynamics as fluid motion under Gaussian density fields. This fluid-inspired framework provides a unified view for balancing editing directionality, structural coherence, and fidelity. We instantiate our approach on StyleGAN2 for text-guided image editing tasks, where preliminary experiments demonstrate its effectiveness in producing semantically accurate, high-quality, and consistent edits compared to conventional latent manipulation methods. Our results suggest that fluid dynamics offers a powerful new paradigm for understanding and guiding latent space transformations in generative models.

1 INTRODUCTION

Text-guided image editing has emerged as a prominent research direction in generative modeling, enabling diverse and fine-grained image manipulations with natural language as high-level semantic conditions. Early approaches combined text embeddings with the StyleGAN style space, either by directly optimizing latent vectors or by learning fixed text-to-direction mappings (Patashnik et al., 2021). While demonstrating the strong capability of text–image alignment in latent manipulation, these methods often rely on per-sample optimization or simple global/linear mappings, making it difficult to jointly ensure semantic precision, consistency, and image fidelity. In parallel, TediGAN (Xia et al., 2021) introduced pipelines based on vision–language similarity learning and instance-level optimization, achieving compelling high-resolution face generation and manipulation. However, such methods typically require additional fine-tuning during inference to preserve identity and structural consistency.

Meanwhile, structured analysis of generative model latent spaces, such as InterFaceGAN (Shen et al., 2020) and GANSpace (Härkönen et al., 2020), revealed interpretable subspaces or principal directions that control attributes like age, expression, and illumination. These geometry-based or statistics-based approaches provided valuable insights for controllable editing, yet they generally treat edits as static, linear, or low-order transformations, leading to overly simplified or stereotyped editing outcomes.

More recent research has explored structured or semantically aligned editing spaces (Lyu et al., 2023a;b) , which construct "delta" feature spaces to connect image-level differences in CLIP space with latent editing directions, enabling more robust training-free or zero-shot generalization. Likewise, CLIP-based inversion and regularization methods (Baykal et al., 2023; Pernuš et al., 2025), integrate text guidance directly into the inversion stage to improve stability and accuracy in multi-attribute editing. Despite these advances in aligning text and image semantics, a fundamental chal-

lenge remains: how to systematically balance the driving force of semantic editing with the constraints that preserve generation consistency and quality during latent manipulation.

In this paper, we introduce a physics-inspired perspective: modeling the latent space of generative models as a high-dimensional Gaussian fluid, and characterizing latent vector evolution during editing through fluid dynamics governed by the Navier–Stokes equations. Within this framework, the movement of latent points is not only driven by text-guided semantic forces but is also constrained by resistances that maintain representational consistency and by a centralizing force imposed by Gaussian density, naturally balancing semantic progression with fidelity at the dynamical level. This fluid-dynamical formulation not only offers a new theoretical perspective for interpreting existing direction-based editing methods, but also directly yields more interpretable optimization update rules—rather than simple interpolation—for text-guided editing in latent space. Finally, we conduct comparative on several face datasets, showing that our method achieves competitive performance in text–image alignment, identity/structure preservation, and generation quality.

2 RELATED WORK

2.1 TEXT-GUIDED IMAGE EDITING

Early GAN-based approaches mainly combined CLIP similarity with latent-space manipulations to achieve intuitive text-conditional edits. For example, StyleCLIP (Patashnik et al., 2021) proposed latent optimization, latent mappers, and global direction techniques to drive StyleGAN editing under text supervision . Extensions of this paradigm further explored domain adaptation and cross-domain transfer using a single pretrained generator: StyleGAN-NADA (Gal et al., 2022) adapted StyleGAN to novel text concepts without paired data , while TediGAN (Xia et al., 2021) introduced multimodal encoders to enable unified text-driven generation and editing, supporting diverse face manipulations. With the advent of diffusion models, a large body of recent work has shifted to leveraging diffusion priors for text-guided editing, typically relying on conditional modeling or inversion. Imagic (Kawar et al., 2023) demonstrated that combining text conditioning with diffusion model inversion and fine-tuning allows for complex, high-fidelity edits of real images. InstructPix2Pix Brooks et al. (2023) trained a diffusion-based editor on text instruction pairs, achieving strong generalization to natural language instructions. Blended Latent Diffusion Avrahami et al. (2023) and its variants developed controllable local/region editing pipelines in latent diffusion space, enabling high-quality interactive edits. Recently, many CLIP- or difference-feature-based methods have emphasized robustness and zero-shot generalization. DeltaEdit (Lyu et al., 2023a) constructed "delta" representations in CLIP feature space to map semantic differences into latent editing directions, supporting flexible text edits without retraining. CLIPInverter (Baykal et al., 2023) incorporated lightweight text-conditioned adapters into GAN inversion pipelines, stabilizing multi-attribute editing for real images. Meanwhile, null-text inversion (Mokady et al., 2023) and other diffusion inversion methods (Wallace et al., 2023; Wang et al., 2024) provided more precise inversion procedures, making text-driven edits of real images more faithful. Locate-and-Forget (Li et al., 2024) improved concept localization and target masking, further enhancing fine-grained text editing accuracy. Despite these advances, most methods still treat editing either as (i) per-sample optimization or (ii) simple linear or global direction updates. As a result, they often face trade-offs between preserving identity/structure and achieving significant semantic changes, sometimes leading to structural distortions or artifacts.

2.2 LATENT SPACE INTERPRETATION AND STRUCTURED MODELING

Understanding the structure of GAN latent spaces has been fundamental for controllable editing. Methods such as InterFaceGAN (Shen et al., 2020) and GANSpace (Härkönen et al., 2020) revealed interpretable directions within latent space, often corresponding to linear subspaces or principal component operations, which enable predictable modifications of attributes such as age, expression, and lighting. Building on these insights, later studies sought to identify more semantically aligned directions (either supervised or unsupervised) and to learn mappings from conditioning signals (e.g., text, attributes) to these directions (Abdal et al., 2021; Shen & Zhou, 2021; Hu et al., 2022). While geometry- and PCA-based analyses provide interpretable control, they typically model editing as static, low-order transformations, and thus cannot capture how latent codes should dynamically evolve under multiple simultaneous constraints.

2.3 PHYSICS- AND DYNAMICS-INSPIRED GENERATIVE MODELING

Another research line models generation as a dynamical transport process between distributions. Diffusion models (Ho et al., 2020) and score-based generative models (Song et al., 2020) formulate data synthesis as time-reversed stochastic dynamics driven by learned score functions, which have become the backbone of state-of-the-art text-to-image generation and editing. Flow Matching Lipman et al. (2022) and Rectified Flow (Esser et al., 2024) propose continuous deterministic or rectified transport formulations, often implemented as neural ODEs or continuous normalizing flows, which train to match distributional transport paths and provide faster sampling trajectories as well as vector-field interpretations. These continuous formulations highlight the benefits of learning and manipulating vector fields in sample or latent spaces, aligning with our view of latent-space editing as a controlled trajectory.

3 METHOD

In this paper, we introduce a physics-inspired framework that models the latent space of generative models as a high-dimensional Gaussian fluid and characterizes the evolution of latent vectors during editing through fluid dynamics governed by the Navier–Stokes equations. Within this formulation, the motion of latent points is driven not only by text-guided semantic forces but also constrained by resistance mechanisms that preserve representational consistency and a centralizing force induced by Gaussian density. This inherently balances semantic progression with structural and perceptual fidelity at a dynamical level. Beyond offering a unified theoretical perspective that interprets existing direction-based editing approaches, our fluid-dynamical model leads to more interpretable and principled optimization rules—going beyond simple interpolation—for text-guided latent space manipulation.

3.1 LATENT SPACE STATISTICAL ANALYSIS

To introduce a physics-inspired modeling perspective, we first investigate the statistical properties of the latent space. We randomly sample 200,000 latent codes from the mapping network of StyleGAN2 and project them into the extended latent space $\mathcal{W}$ using the pretrained generator.

Previous studies (Härkönen et al., 2020; Shen et al., 2020; Shen & Zhou, 2021) have demonstrated that certain directions in the latent space are closely associated with semantic attributes of images, such as age, pose, expression, or illumination. The principal components extracted through Principal Component Analysis (PCA) not only exhibit strong correlations with such semantic factors but also possess geometric orthogonality, naturally satisfying the conditions for forming a basis of the space. This ensures that any latent vector can be uniquely decomposed in this basis, thereby providing a solid theoretical foundation for modeling the latent space as a multidimensional fluid composed of independent variables.

We perform PCA on all samples to obtain a set of mutually orthogonal principal component directions $\{u_i\}_{i=1}^{d}$. We then project the samples onto each principal component direction:

$$z_i = w \cdot u_i, \quad i = 1, 2, \ldots, d \tag{1}$$

We then conduct Kolmogorov–Smirnov and Shapiro–Wilk normality tests on these projections. The results show that most projections closely follow a Gaussian distribution:

$$z_i \sim \mathcal{N}(0, \sigma_i^2), \quad i = 1, 2, \ldots, d \tag{2}$$

This result indicates that, under the PCA basis, the latent space can be approximated as a set of independent Gaussian variables.

3.2 SEMANTIC TARGET ESTIMATION VIA PRINCIPAL COMPONENT COEFFICIENTS

Building on the statistical decomposition of the latent space, we model the editing dynamics along the orthogonal axes defined by principal components. Instead of treating semantic directions as entangled or learned in an implicit embedding space, we directly estimate coefficients corresponding to each principal component (PC). These coefficients naturally serve as the "semantic target" that govern the transformation of a latent code.

Formally, given a latent code w and a set of principal component vectors $\{u_i\}_{i=1}^{d}$, we predict a coefficient vector $\alpha = [\alpha_1, \alpha_2, \ldots, \alpha_d]$. The edited latent code is then expressed as

$$w' = w + \sum_{i=1}^{d} \alpha_i u_i \tag{3}$$

where α_i directly determines the magnitude of displacement along its corresponding semantic direction u_i. This design offers more controllable and interpretable editing, since forces along different PCs can be independently adjusted or constrained to achieve desired editing effects.

3.3 Fluid Dynamics-inspired Optimization

In the previous sections, we characterized the magnitude of the editing driving force through the variation coefficients of principal component directions. However, merely relying on linear weighting to balance semantic driving force and consistency resistance often fails to adapt adequately to the dynamic requirements of different editing tasks. Most existing editing trade-off schemes are essentially linear, where editing directions are combined with hyperparameters or fixed ratios. This approach is equivalent to linearly interpolating between the original and target distributions. Consequently, interpolated samples may simultaneously deviate from the high-density regions of both distributions, leading to degraded generation quality.

To address this limitation, we propose to model the latent space as a multivariate fluid and optimize editing trajectories under the framework of fluid dynamics. Our idea originates from the observation that in continuous fluid systems, the velocity field can be naturally described by the Navier–Stokes equation (Tao, 2016), given the external forces and fluid properties such as density and viscosity. This formulation inherently provides a principled way to integrate external semantic forces with central constraints arising from data density. In particular, it allows the editing trajectory to dynamically balance semantic alignment, consistency preservation, and generation quality, rather than relying on rigid linear weighting schemes.

Simplification Given the Gaussian-like property of latent projections along principal components, we interpret the entire latent space as a multidimensional Gaussian fluid, where each principal direction corresponds to an independent Gaussian density. This perspective allows us to borrow tools from fluid dynamics to model latent code transformations as trajectories within such a fluid.

We begin from the Navier–Stokes equation, which governs the motion of particles in a fluid:

$$\rho\left(\frac{\partial v}{\partial t} + (v \cdot \nabla)v\right) = -\nabla p + \mu \nabla^2 v + f \tag{4}$$

where ρ is the fluid density, v is the velocity field, p is the pressure, μ is the dynamic viscosity, and f represents external forces acting on the fluid.

Since we are interested in semantic editing dynamics rather than full turbulence modeling, we impose the following assumption: we assume constant velocity within each editing step, thereby neglecting acceleration, inertia, and viscous drag terms. As a result of our simplification,the particle motion reduces to a balance between density-gradient forces and external semantic forces:

$$\mathbf{v} = \frac{-\nabla \Phi(\mathbf{x}) + \mathbf{F}_{\text{ext}}}{\eta} \tag{5}$$

where η is a learnable viscosity parameter, $\Phi(\mathbf{x})$ denotes the density-gradient force, and $\mathbf{F}_{\text{ext}}$ represents the external force.

Density-gradient force (Central Constraint) Since we assume that the latent follows a Gaussian distribution along each principal component, i.e.

$$x_i \sim \mathcal{N}(\mu, \sigma_i^2) \tag{6}$$

The density at point $\mathbf{x}$ is given by

$$\rho(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \tag{7}$$

We define the central constraint as the gradient of negative log-density:

$$\nabla\Phi(x) = \nabla(-\log\rho(x)) = \frac{(x-\mu)}{\sigma^2} \tag{8}$$

This force naturally drives latent codes back toward the Gaussian center, ensuring stability and preventing unrealistic drifts.

External Forces (Semantic Drive and Consistency Resistance) In addition to the center constraint, editing requires external forces that balance semantic alignment with fidelity preservation. We decompose the external force into two components: a semantic driving force that pushes the latent code toward the target direction, and a resistance force that preserves consistency with the original image.

Semantic driving force force pushes the latent code toward the target semantic direction specified by the target. Formally, it is expressed as

$$\mathbf{F}_{sem} = \nabla(-\log\rho_{target}(x)) \tag{9}$$

where $\rho_{target}(x)$ measures the semantic consistency between the current latent code and the target. This term ensures that the editing trajectory converges toward the desired semantics.

To avoid excessive deviation that could harm identity or structure, we introduce a force that encourages proximity to the original latent:

$$\mathbf{F}_{con} = -\nabla(-\log\rho_{orig}(x)) \tag{10}$$

where $\rho_{orig}(x)$ measures the similarity between the current latent and the original latent. This term acts as a resistance, preventing over-editing.

The overall external force is then given by:

$$\mathbf{F}_{ext} = \mathbf{F}_{sem} + \mathbf{F}_{con} \tag{11}$$

Learnable Viscosity and Integration The damping factor η is predicted by a neural network conditioned on both the latent code and the text prompt, enabling adaptive control of editing dynamics.

The resulting velocity is then:

$$\mathbf{v} = \frac{-\nabla\Phi(\mathbf{x}) + \mathbf{F}_{sem} + \mathbf{F}_{con}}{\eta} \tag{12}$$

Finally, the updated latent code after one editing step is obtained by time integration within the normalized interval $[0, 1]$:

$$x^* = x + \int_0^1 \mathbf{v}(t)dt \tag{13}$$

This formulation ensures that editing outcomes reflect a balanced trade-off among semantic fidelity, consistency, and image realism.

3.4 Loss Functions

We use CLIP to encode text conditions, and define the training loss as the weighted sum of CLIP text–image similarity loss, LPIPS perceptual loss, and ArcFace identity loss. The weights of all three components are set to 1.0. Our training procedure consists of two stages, corresponding to the principal component coefficient prediction in Section. 3.2 and the latent fluid dynamics optimization in Section. 3.3. Accordingly, we design two groups of loss functions tailored for each stage.

Stage 1: Principal Component Coefficient Prediction. The goal of this stage is to predict the coefficients α along the principal component axes while ensuring semantic consistency and high-fidelity reconstruction. The loss function integrates the following components: 1) Reconstruction Loss. We adopt an L1 loss to enforce pixel-level consistency between the reconstructed and input images. To further enhance perceptual quality and identity preservation, we incorporate the LPIPS perceptual loss (Zhang et al., 2018) and the ArcFace identity loss (Deng et al., 2019). 2) Semantic

Alignment Loss. We leverage CLIP image–text similarity (Radford et al., 2021) to ensure that the generated edits conform to the target text prompts. 3) Coefficient Regularization. An L2 regularization on the predicted coefficients α prevents over-editing and ensures stable and interpretable editing directions.

The overall objective for Stage 1 is formulated as:

$$\mathcal{L}_{stage1} = \lambda_{L1}\mathcal{L}_{L1} + \lambda_{LPIPS}\mathcal{L}_{LPIPS} + \lambda_{id}\mathcal{L}_{id} + \lambda_{CLIP}\mathcal{L}_{CLIP} + \lambda_{reg}\|\alpha\|_2^2 \tag{14}$$

Stage 2: Latent Fluid Dynamics Optimization. In the second stage, we focus on refining the editing trajectory using the latent fluid dynamics optimization framework. The loss function includes: 1) Semantic Alignment Loss. CLIP image–text similarity is again employed to enforce alignment with the target text conditions. 2) Latent Regularization. We introduce an L2 regularization on the density discrepancy in the latent space to prevent deviations from the Gaussian prior, ensuring that the optimization remains stable within the fluid model. 3) Fidelity Losses. Because this optimization base on the Sec. 3.2 during training, we also include the LPIPS perceptual loss and ArcFace identity loss to maintain high-fidelity reconstruction and identity preservation. The overall objective for Stage 2 is given by:

$$\mathcal{L}_{stage2} = \lambda_{CLIP}\mathcal{L}_{CLIP} + \lambda_{latent}\mathcal{L}_{latent} + \lambda_{LPIPS}\mathcal{L}_{LPIPS} + \lambda_{id}\mathcal{L}_{id} \tag{15}$$

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

We conduct experiments on face image editing to validate the effectiveness of our proposed Gaussian fluid-based latent trajectory optimization framework.

Datasets and pretrained models Our method is trained on the CelebAMask-HQ (Xia et al., 2021) and evaluated on both CelebAMask-HQ and the FFHQ Karras et al. (2019). For quantitative evaluation, we randomly select text prompts from the CelebA dataset and perform text-guided editing tasks. We adopt the e4e (Tov et al., 2021) encoder as the inversion module to project input images into the latent space, and employ a pretrained StyleGAN2 (Karras et al., 2020) generator as the backbone synthesis network.

Implementation details In practice, the fluid dynamics formulation introduced in eq. 12 requires the viscosity parameter η to balance the contributions of semantic driving force $\mathbf{F}_{sem}$ and consistency resistance $\mathbf{F}_{con}$. To better adapt to different editing conditions, we implement this balance as a linear combination $\lambda_{sem}\mathbf{F}_{sem} + \lambda_{con}\mathbf{F}_{con}$, where λ_{sem} and λ_{con} are learnable weights predicted by a neural network conditioned on both the latent code and the text prompt. This design allows the model to dynamically adjust the emphasis on semantic alignment versus consistency preservation based on the specific editing context.

Baselines and evaluation metrics We compare our approach with four representative text-guided image editing methods: TediGAN (Xia et al., 2021), StyleCLIP (Patashnik et al., 2021), CLIPInvert (Baykal et al., 2023), and DeltaEdit (Lyu et al., 2023a). These baselines cover direction-based latent editing, inversion-based methods, and recent optimization-driven frameworks, providing a comprehensive benchmark.

To evaluate editing quality, we adopt four metrics: 1) CLIP-I: CLIP-based similarity between edited and source images, reflecting content preservation. 2) CLIP-T: CLIP-based similarity between edited images and target text, measuring semantic alignment. 3) DINOScore: similarity computed by DINOv2, serving as an additional perceptual metric. 4) AestheticsScore: aesthetic quality score introduced by LAION, evaluating overall visual appeal.

4.2 RESULTS AND ANALYSIS

4.2.1 QUANTITATIVE RESULTS

We present quantitative comparisons on the CelebAMask-HQ and FFHQ datasets in Tables 1 and 2, respectively. Several clear trends can be observed.

Table 1: Quantitative comparison on CelebAMask-HQ dataset. Best results are highlighted in **bold**.

Method	CLIP-I ↑	CLIP-T ↑	DINOScore ↑	Aesthetics ↑
TediGAN	0.633	**0.343**	**0.780**	5.096
StyleCLIP	0.806	0.197	0.736	**5.352**
CLIPInverter	0.729	0.279	0.619	5.157
DeltaEdit	0.781	0.216	0.697	5.324
Ours (CoeffPredictor)	0.801	0.236	0.736	5.296
Ours (LFDO)	**0.827**	0.222	0.766	5.342

On CelebAMask-HQ, our PCA-force variant already achieves competitive performance, with balanced scores across CLIP-I, CLIP-T, DINOScore, and Aesthetics. By incorporating the proposed LFDO, our method further improves CLIP-I from 0.801 to 0.827 and DINOScore from 0.736 to 0.766, while maintaining a strong Aesthetics score (5.342). Compared to baselines, TediGAN excels in CLIP-T (0.343) and DINOScore (0.780) but suffers from lower identity preservation (CLIP-I: 0.633). Notably, as an inference-time optimization method, TediGAN requires significant computational time per edit, limiting its practical applicability. StyleCLIP achieves high perceptual quality (Aesthetics: 5.352) but struggles with semantic alignment (CLIP-T: 0.197). Moreover, StyleCLIP relies on paired training data for specific edits, restricting its generalization to arbitrary text prompts. CLIPInverter and DeltaEdit provide more balanced results but underperform our method in both semantic alignment and visual similarity. These results suggest that our approach achieves a better trade-off among fidelity, semantic accuracy, and image quality.

Table 2: Quantitative comparison on FFHQ dataset. Best results are highlighted in **bold**.

Method	CLIP-I ↑	CLIP-T ↑	DINOScore ↑	Aesthetics ↑
TediGAN	0.627	**0.329**	**0.735**	4.905
CLIPInverter	0.718	0.254	0.563	**5.130**
DeltaEdit	0.847	0.205	0.676	5.087
Ours (CoeffPredictor)	0.823	0.227	0.657	5.032
Ours (LFDO)	**0.858**	0.211	0.695	5.035

On FFHQ, a similar trend is observed. Our PCA-force variant shows robust performance, and the fluid optimization further boosts CLIP-I (from 0.823 to 0.858) and DINOScore (from 0.657 to 0.695), surpassing all baselines. TediGAN maintains strong CLIP-T (0.329) but performs poorly in identity and aesthetics, with the additional drawback of slow inference due to its optimization-based nature. DeltaEdit is competitive in CLIP-I (0.847) but lags behind in semantic alignment. In contrast, our method consistently balances semantic alignment, fidelity, and visual quality across datasets.

These results highlight the advantage of modeling editing trajectories as fluid dynamics. Unlike inference-optimization approaches (e.g., TediGAN) or pair-dependent methods (e.g., StyleCLIP), our framework produces stable improvements across datasets and metrics, demonstrating both generality and robustness.

4.2.2 Qualitative Results

We present qualitative comparisons to further highlight the advantages of our approach. Figure. 1 showcases the editing performance of different baselines. The leftmost column shows the original images, followed by their inverted reconstructions. Each subsequent row corresponds to one editing method under various text prompts, with prompts placed below each column and model names listed along the rightmost column. We observe that baseline methods often introduce undesired modifications unrelated to the target prompt. In contrast, our Delta Predictor effectively mitigates such spurious changes and maintains stable compositional generation. We attribute this to the disentanglement effect induced by linear combinations along principal component directions. Furthermore, our fluid-dynamics optimization suppresses overly exaggerated modifications, reduces semantic-irrelevant edits, and preserves overall visual quality.

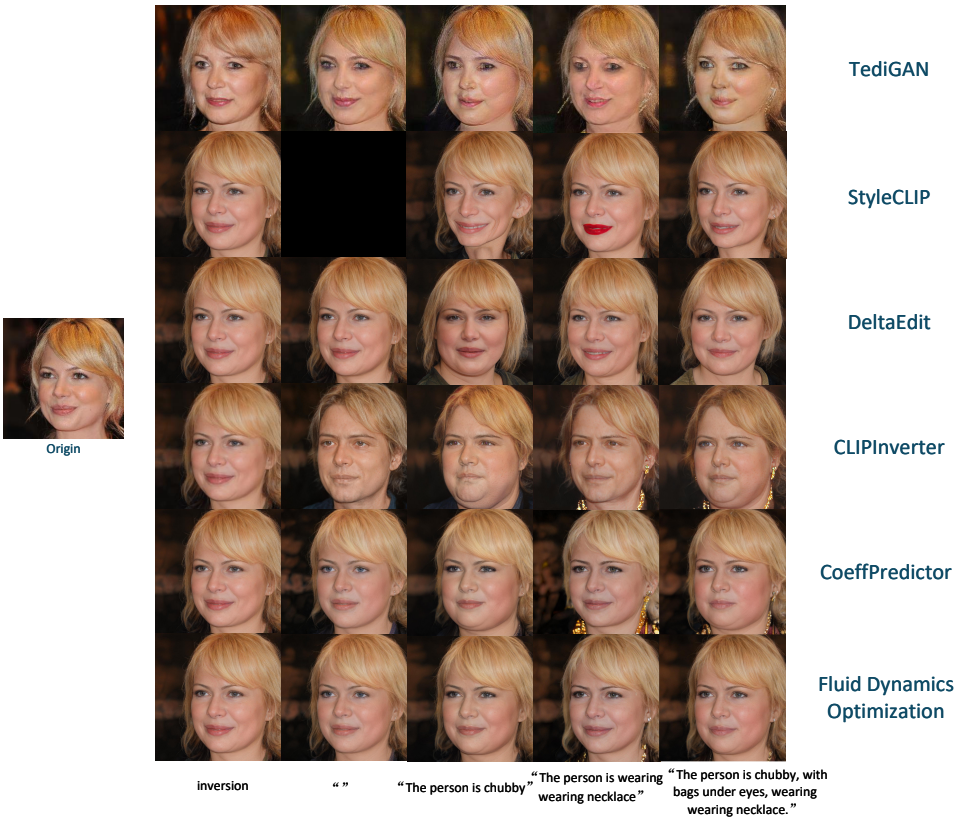

Figure 1: Qualitative comparison of text-guided image editing. The leftmost column displays the original images. In the generated results, the first column corresponds to the inverted reconstructions, while the subsequent columns present the edited outcomes under different text prompts. The text prompts are provided below each column, and the method names are indicated on the right-hand side. Our method corresponds to the last two rows. Since StyleCLIP does not support empty text, its empty text image is ignored.

To further illustrate the benefits of fluid optimization, Figure. 2 compares the outputs of Coefficient Predictor (top row) and its optimized version (bottom row) under the same set of prompts. While Delta Predictor occasionally produces unintended edits—for example, generating long hair when prompted with “straight hair,” or altering expressions when asked for “heavy makeup”—the optimized results yield milder edits that remain semantically faithful, significantly reducing irrelevant changes.

Figure 2: Comparison between Coefficient Predictor (top row) and its fluid-optimized version (bottom row) under the same text prompts. The optimized results exhibit more controlled and semantically faithful edits, reducing unintended modifications.

Finally, Figure. 3 demonstrates the versatility of our method by applying latent fluid-dynamics optimization to CLIPInverter. Since CLIPInverter directly adds editing directions in the w-space, these directions can be projected onto principal component axes via:

$$\alpha = \Delta w \cdot U^{-1} U^{-1} = U^T \tag{16}$$

where U is a square matrix formed by horizontally stacking the principal component vectors, α is the coefficient vector introduced in Section. 3.2, and δw represents the editing direction in the w-space. Since the principal component matrix is orthonormal, its inverse equals its transpose. Subsequently, the coefficients α can be used as the semantic driving force described in our optimization framework.

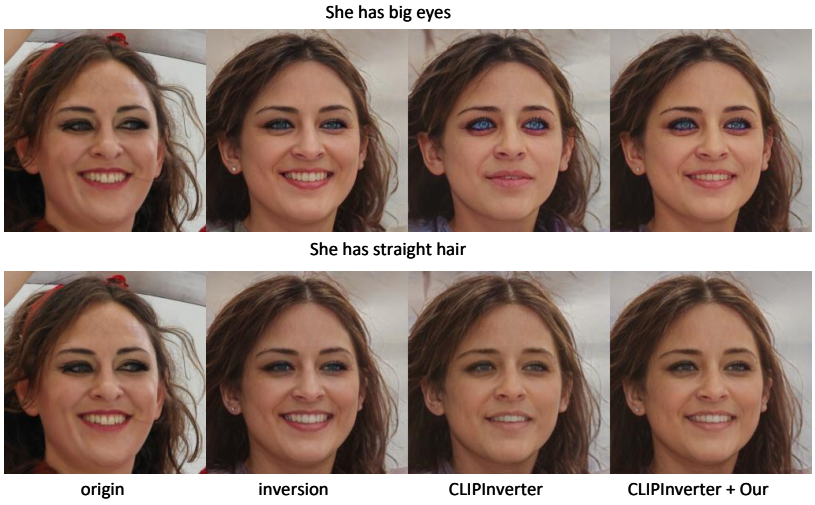

Figure 3: Applying fluid-dynamics-based optimization to CLIPInverter. The first column is the original image, the second column is the inversion of CLIPInverter, the third column is the edited result of CLIPInverter, and the last column is the result of the third column after our optimization method. Our method uses the inversion as the optimization benchmark.The optimized results better preserve identity and reduce excessive edits.

As shown in Figure 3, applying our fluid-dynamics-based optimization method to CLIPInverter effectively suppresses excessive edits and better preserves identity information, demonstrating the broad applicability of our proposed approach.

4.3 Ablation Study

Due to time constraints and the current limitations of our training strategy, we were unable to conduct a comprehensive set of ablation studies. Partial experimental details and preliminary results are provided in the Appendix. We plan to further validate our method through more extensive ablations in future work and will include the corresponding results in an updated version of the Appendix.

4.4 Conclusion and Discussion

In this paper, we explore the latent space of pretrained generative models and propose a principled framework for text-guided image editing. Specifically, we leverage principal component directions as an interpretable basis and design a coefficient-prediction scheme to achieve disentangled and controllable edits. Furthermore, to address the challenges of over-editing and irrelevant semantic changes, we introduce a latent fluid dynamics optimization method, which dynamically balances semantic accuracy, identity preservation, and visual quality.

Although our current experiments mainly simulate degradation scenarios via linear scaling of Δ_{latent}, the proposed framework already demonstrates clear improvements over existing baselines. In particular, principal component modeling provides strong interpretability and controllability, while fluid-inspired optimization offers a natural way to regularize editing trajectories. We believe that extending this framework to nonlinear degradation and more complex editing tasks will further validate its potential. As part of our future work, we plan to refine our experimental design to include nonlinear distortions and richer editing dynamics, providing a more rigorous evaluation of our optimization approach.

References

Rameen Abdal, Peihao Zhu, Niloy J Mitra, and Peter Wonka. Styleflow: Attribute-conditioned exploration of stylegan-generated images using conditional continuous normalizing flows. *ACM Transactions on Graphics (ToG)*, 40(3):1–21, 2021.

Omri Avrahami, Ohad Fried, and Dani Lischinski. Blended latent diffusion. *ACM transactions on graphics (TOG)*, 42(4):1–11, 2023.

Ahmet Canberk Baykal, Abdul Basit Anees, Duygu Ceylan, Erkut Erdem, Aykut Erdem, and Deniz Yuret. Clip-guided stylegan inversion for text-driven real image editing. *ACM Transactions on Graphics*, 42(5):1–18, 2023.

Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 18392–18402, 2023.

Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4690–4699, 2019.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

Rinon Gal, Or Patashnik, Haggai Maron, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain adaptation of image generators. *ACM Transactions on Graphics (TOG)*, 41(4):1–13, 2022.

Erik Härkönen, Aaron Hertzmann, Jaakko Lehtinen, and Sylvain Paris. Ganspace: Discovering interpretable gan controls. *Advances in neural information processing systems*, 33:9841–9850, 2020.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Xueqi Hu, Qiusheng Huang, Zhengyi Shi, Siyuan Li, Changxin Gao, Li Sun, and Qingli Li. Style transformer for image inversion and editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11337–11346, June 2022.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4401–4410, 2019.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8110–8119, 2020.

Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6007–6017, 2023.

Jia Li, Lijie Hu, Zhixian He, Jingfeng Zhang, Tianhang Zheng, and Di Wang. Text guided image editing with automatic concept locating and forgetting. *arXiv preprint arXiv:2405.19708*, 2024.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Yueming Lyu, Tianwei Lin, Fu Li, Dongliang He, Jing Dong, and Tieniu Tan. Deltaedit: Exploring text-free training for text-driven image manipulation. *arXiv preprint arXiv:2303.06285*, 2023a.

Yueming Lyu, Kang Zhao, Bo Peng, Yue Jiang, Yingya Zhang, and Jing Dong. Deltaspace: A semantic-aligned feature space for flexible text-guided image editing. *arXiv preprint arXiv:2310.08785*, 2023b.

Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6038–6047, 2023.

Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 2085–2094, 2021.

Martin Pernuš, Clinton Fookes, Vitomir Štruc, and Simon Dobrišek. Fice: Text-conditioned fashion-image editing with guided gan inversion. *Pattern Recognition*, 158:111022, 2025.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Yujun Shen and Bolei Zhou. Closed-form factorization of latent semantics in gans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 1532–1540, 2021.

Yujun Shen, Ceyuan Yang, Xiaoou Tang, and Bolei Zhou. Interfacegan: Interpreting the disentangled face representation learned by gans. *IEEE transactions on pattern analysis and machine intelligence*, 44(4):2004–2018, 2020.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

Terence Tao. Finite time blowup for an averaged three-dimensional navier-stokes equation. *Journal of the American Mathematical Society*, 29(3):601–674, 2016.

Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. Designing an encoder for stylegan image manipulation. *ACM Transactions on Graphics (TOG)*, 40(4):1–14, 2021.

Bram Wallace, Akash Gokul, and Nikhil Naik. Edict: Exact diffusion inversion via coupled transformations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22532–22541, 2023.

Fangyikang Wang, Hubery Yin, Yue-Jiang Dong, Huminhao Zhu, Hanbin Zhao, Hui Qian, Chen Li, et al. Belm: Bidirectional explicit linear multi-step sampler for exact inversion in diffusion models. *Advances in Neural Information Processing Systems*, 37:46118–46159, 2024.

Weihao Xia, Yujiu Yang, Jing-Hao Xue, and Baoyuan Wu. Tedigan: Text-guided diverse face image generation and manipulation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2256–2265, 2021.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 586–595, 2018.

APPENDIX

A SUPPLEMENTARY ANALYSIS OF LATENT SPACE PROPERTIES (RELATED TO SECTION. 3.1)

To further investigate the statistical properties of the latent space, we conducted additional experiments on StyleGAN2. Specifically, we randomly sampled 200k data points from the z space of the pretrained model and mapped them into the latent space. We then performed Principal Component Analysis (PCA) to compute the mean, covariance, and principal component directions of the distribution.

From the 200k data points, we randomly sampled subsets of size 100 and 500, projecting them along each principal direction. We subsequently applied two standard normality tests: the Kolmogorov–Smirnov (K-S) test and the Shapiro–Wilk test.

K-S test results With 100 samples, all p-values were greater than 0.05, and the test statistic D remained small, suggesting normality. With 500 samples, 6 dimensions showed p-values smaller than 0.05, almost all of which correspond to the last principal components with relatively small variance.

Shapiro–Wilk test results The W statistics were consistently close to 1. For 100 samples, 50 dimensions yielded p-values below 0.05, and for 500 samples, this number increased to 90. Again, these violations were mainly concentrated in principal directions with low variance contribution.

In our experiments, we worked in the w^+ space, where we averaged n latent codes before PCA. This averaging step may amplify cumulative errors. Nevertheless, the vast majority of dimensions still approximately follow a normal distribution. Based on these findings, we conclude that the StyleGAN2 latent space can be reasonably modeled as a multivariate Gaussian distribution in the PCA basis.

For illustration, we also randomly sampled 2,000 data points and examined one particular dimension. Figure. 4 shows its Q-Q plot, with the Shapiro–Wilk p-value equal to 0.579. Figure. 5 shows the corresponding Gaussian fit, with a K-S test p-value of 0.831. These results further support our assumption of approximate Gaussianity in most latent dimensions.

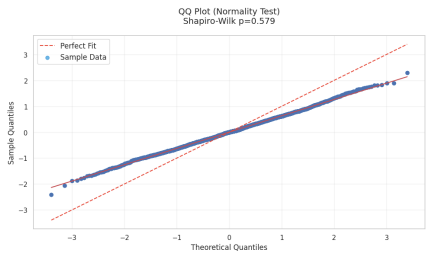

Figure 4: Q-Q plot of one principal component direction, showing approximate normality.

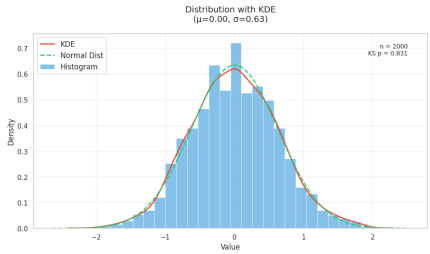

Figure 5: Histogram and Gaussian fit of the same principal component direction, with K-S test p-value of 0.831.

B Additional Results

B.1 Additional Results on Principal Component Coefficients

We provide additional details and results regarding the experiments on principal component coefficients.

Experimental details Our coefficient prediction network adopts a U-Net architecture with cross-attention. The learning rate is set to 0.001. We use a weighted loss with the following coefficients: $\lambda_{\text{clip}} = 3.0$, $\lambda_{\text{lpips}} = 1.0$, $\lambda_{\text{id}} = 1.0$, $\lambda_{L1} = 1.0$, and $\lambda_{\text{reg}} = 0.05$.

Effect of combining principal components In the first visualization (Fig. **??**), each row corresponds to editing results using a different number of principal components predicted by our coefficient network. The editing prompt is displayed at the top of the figure. From top to bottom, the rows correspond to using the top $[1, 2, 5, 10, 512]$ principal components, respectively. Within each row, the columns represent increasing editing coefficients $[1, \ldots, 10]$, where the editing direction is given by scaling the linear combination of the principal components.

The results demonstrate that the ability to perform composite edits improves as more principal components are included. For example, using only a single principal direction fails to achieve multi-attribute edits, and using two directions is insufficient for structural transformations such as body posture. As more components are added, the editing becomes more sensitive to the editing coefficients, indicating that multiple directions jointly contribute to controlling complex attributes. This validates our hypothesis that while principal directions are statistically independent, a single semantic attribute is often entangled across multiple directions. Thus, editing with only one direction is more likely to introduce undesired semantic changes, whereas combining multiple components can mitigate such side effects.

Effect of varying editing coefficients In the second visualization (Fig. 7), we analyze how different coefficient magnitudes affect the results. From left to right, the editing coefficients are set to $[-2.5, -2.0, -1.5, -1.0, -0.5, 0.0, 0.5, 1.0, 1.5, 2.0, 2.5]$. The first row shows the results of CLIP-Inverter, while the second row presents our method.

We observe that our approach provides greater flexibility and semantic alignment across a wide range of coefficient values. When the editing coefficients are large in magnitude, our method better preserves attributes unrelated to the prompt, avoiding excessive or unintended changes. This highlights the robustness of our coefficient-based editing scheme in maintaining disentanglement while enabling precise control over the editing strength.

B.2 Additional Results of Latent Fluid Dynamics Optimization

Our proposed latent fluid dynamics optimization builds upon the Navier–Stokes formulation to refine editing trajectories. Due to the current stage of implementation and training, we have not yet completed comprehensive experiments for this section. Preliminary results are being actively developed, and we plan to extend this subsection with detailed visualizations and quantitative analysis in future revisions of this work.

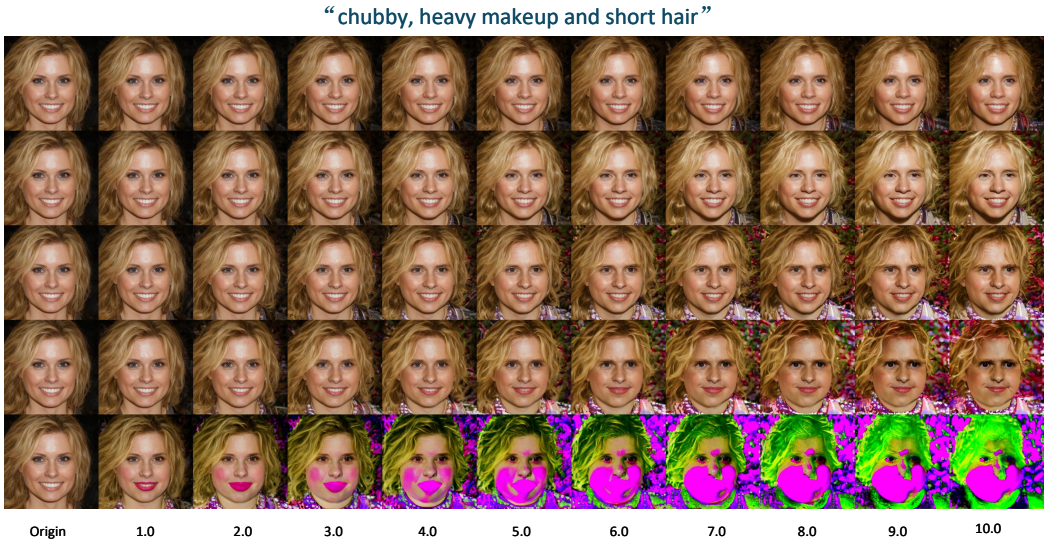

Figure 6: Editing results using different numbers of principal components. Each row corresponds to a different number of components, and each column represents increasing editing coefficients. The prompt is ”A person with heavy makeup and straight hair”.

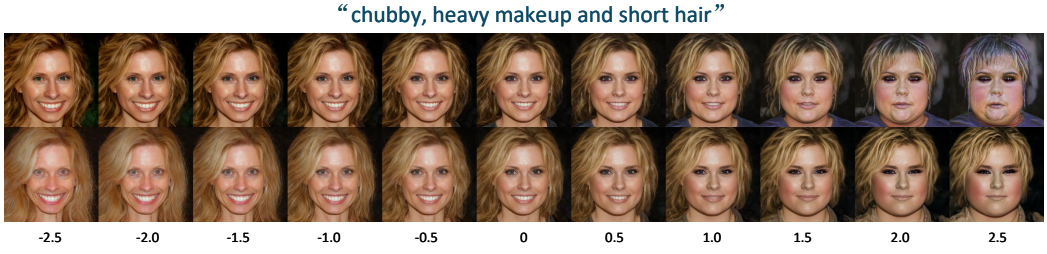

Figure 7: Editing results with varying coefficients. The first row shows CLIPInverter results, and the second row shows our method’s results. The prompt is ”A person with heavy makeup and straight hair”.

DISCLOSURE OF LLM ASSISTANCE

We acknowledge that large language models (LLMs) were employed during the preparation of this manuscript. Specifically, the models were used to help refine and polish the textual descriptions of our methods and experiments, with the goal of improving clarity and readability. All technical ideas, formulations, experiments, and results presented in this paper are solely the work of the authors. The use of LLMs was limited to language assistance and did not influence the scientific contributions of this research.