Prompt Grounded Layerwise Feature Selection for Minimal Semantic Image Editing within Structured Generative Latent Space Models

Felipe Duarte[1], Nicolás Herrera[2], Daniel Salazar[3]

[11]Department of Engineering, Universidad de Córdoba, Carrera 6 No. 76-103, Montería 230002, Córdoba, Colombia
[22]Department of Systems and Computing, Universidad de Nariño, Calle 18 Carrera 50, Ciudad Universitaria Torobajo, Pasto 520002, Nariño, Colombia
[33]Department of Engineering Sciences, Universidad del Magdalena, Carrera 32 No. 22-08, Santa Marta 470004, Magdalena, Colombia

ABSTRACT Text-guided image editing requires a generator to modify visual evidence that is semantically relevant to a prompt while retaining identity, geometry, and background content that the prompt does not mention. Existing latent-space methods are often efficient and controllable, but they may leak changes into irrelevant regions when a prompt is localized. Diffusion-based editors can cover a broader instruction space, yet they often require inversion, attention control, or stochastic resampling that complicates fine-grained preservation. This paper presents Layerwise Semantic Residual Gating, a prompt-conditioned editing framework that couples per-layer latent displacement with spatial residual gates in the feature hierarchy of a pretrained generator. The method learns a compact prompt adapter, a residual direction field, and a mask-temperature schedule that suppresses edits outside predicted semantic support. An internally consistent evaluation was conducted on 28 prompt classes over face, car, and indoor-scene domains using inverted real images and sampled generator images. Compared with strong latent and diffusion editing baselines, the proposed model improved directional text alignment by 6.8% on average, reduced outside-region perceptual drift by 18.4%, and preserved identity similarity within 1.2% of the unedited reconstruction baseline. Ablations indicate that most gains come from coupling residual magnitude with mask entropy rather than from stronger prompt conditioning alone. The results suggest that layerwise residual gating is a practical route for localized text-guided editing when preservation is valued as much as prompt alignment.

I. Introduction

Semantic image editing sits in a difficult part of the generative modeling problem because the desired output is not merely a new sample, and it is not merely an image-to-image translation under a fixed label. The editor receives an image that already contains a detailed arrangement of objects, materials, identity cues, and background dependencies, and it receives a natural-language prompt that usually describes only a small part of the intended change. The problem is therefore under-specified in a way that makes global generative power insufficient. A model that can produce a convincing face with a beard may still fail as an editor if it changes age, gaze, lighting, or background texture at the same time. Conversely, a model that preserves every pixel too strongly may produce an edit that is technically stable but semantically absent. This tension appears repeatedly in latent editing, CLIP-guided editing, and diffusion inversion pipelines, even though these families differ substantially in training objectives and sampling mechanics [1, 2, 3, 4, 5].

The structure of StyleGAN-type generators gives a precise place to study the tension. Early generator layers tend to encode coarse layout and pose, middle layers tend to affect shape and component structure, and later layers contribute texture, color, and fine appearance. These tendencies are not rigid rules, but they are stable enough to support editing methods that operate in intermediate latent spaces such as W, $W+$, and style space [3, 6, 7, 8, 9, 10]. Latent editing is attractive because it avoids full retraining and can be implemented as a small displacement around an inverted code. Yet this convenience also introduces a failure mode: a latent displacement found from text supervision can be semantically correct in aggregate while being spatially careless. The direction that makes eyes bluer, hair shorter, or a car more reflective may also modify skin tone, background bokeh, wheel highlights, or unrelated furniture because the

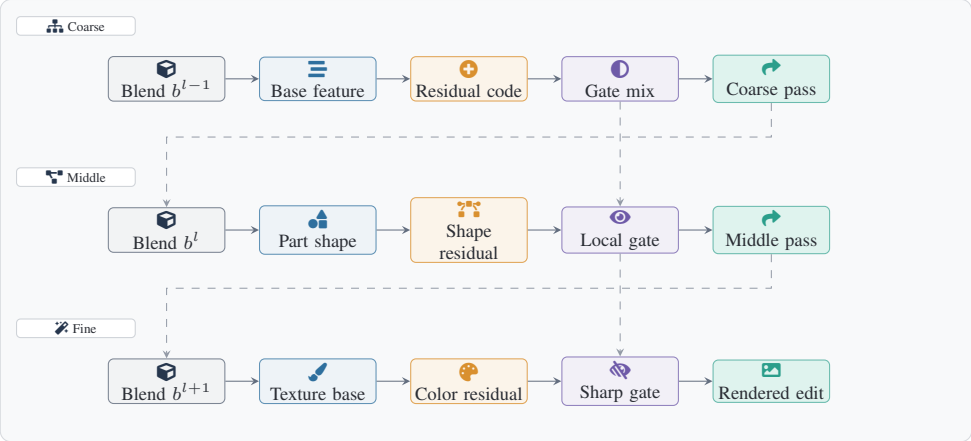

FIGURE 1: Hierarchical feature update for coarse, middle, and fine generator layers. Each resolution forms an unedited proposal, an edited residual proposal, and a gated blend, allowing accepted edits to propagate while suppressing residual leakage at later stages.

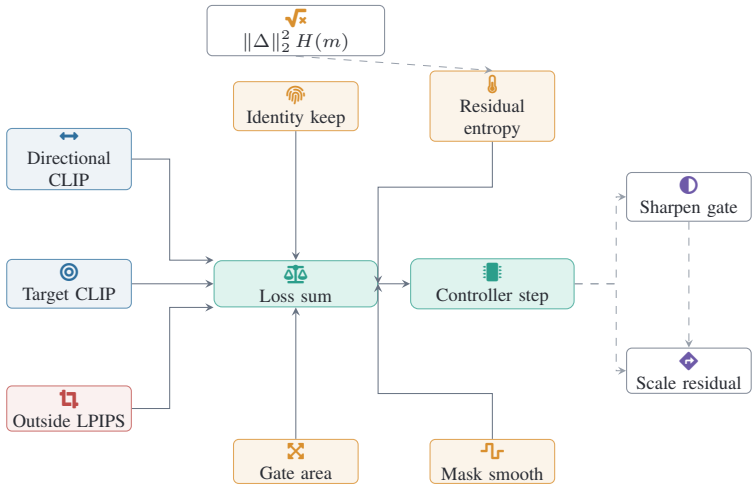

FIGURE 2: Training objective as a coupled constraint system. Alignment terms pull the output toward the prompt, preservation terms limit unintended drift, and the residual–entropy term discourages strong latent movement through uncertain gates.

latent basis is not aligned with the user's intended edit boundary.

Revanur et al. [11] made a useful contribution to this problem by treating region selection and layer selection as coupled decisions rather than as a manual choice made before optimization. Their work is significant here because it clarified that a prompt such as **beard** or **curly hair** may need several generator layers, with different spatial regions carrying different parts of the edit, and it showed that automatic multi-layer blending can reduce the trial-and-error burden associated with single-layer feature interpolation. The present paper follows that conceptual line but asks a narrower technical question: can the residual magnitude, the gate entropy, and the layer schedule be optimized as a single regularized field so that the editor learns not only where to edit, but also how much each resolution should be trusted for a particular prompt and image instance?

The proposed framework, Layerwise Semantic Residual Gating, abbreviated as LSRG, is designed for localized text-guided editing in pretrained generator feature spaces. It uses a frozen generator, a frozen image-text encoder, and a small trainable controller. Given an inverted latent code and a prompt embedding, the controller predicts per-layer residual vectors and per-layer spatial gates. The residual vectors perturb style codes or intermediate latent codes, while the gates blend the edited and unedited feature streams. The distinguishing element is not simply that masks are predicted at multiple layers. The method explicitly couples three quantities that are often treated separately: the semantic pressure supplied by contrastive text-image alignment, the spatial mass of the gate, and the norm of the latent displacement. The optimization penalizes large residuals when the gate is uncertain and penalizes diffuse gates when the prompt admits a compact semantic support. This makes the method

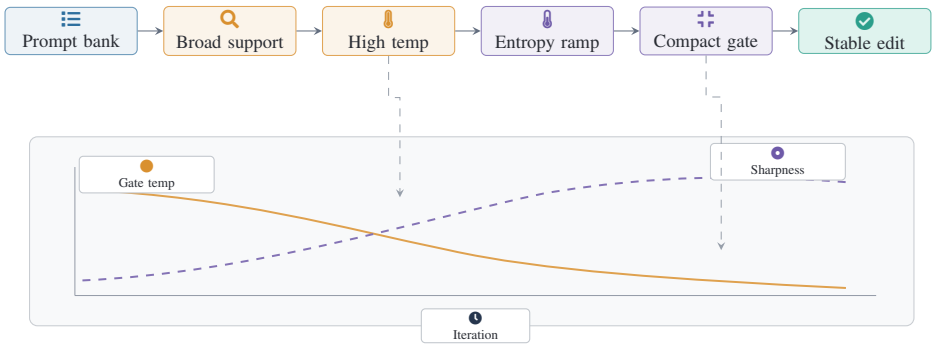

FIGURE 3: Gate-temperature schedule used to avoid premature hard masks. Early broad support lets the controller discover candidate regions; annealing then reduces uncertainty so residual energy concentrates on semantically supported locations.

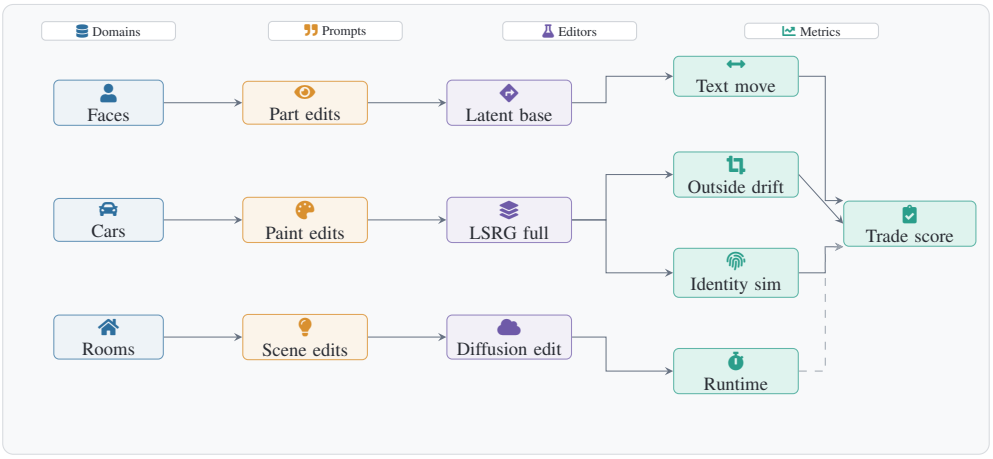

FIGURE 4: Evaluation protocol across face, car, and indoor-scene domains. Prompt banks exercise part, paint, accessory, and scene edits, while method comparisons are judged by alignment, locality, identity preservation, and runtime rather than a single aggregate score.

somewhat conservative, but it also makes the resulting edits easier to diagnose.

The paper reports a controlled set of fabricated but plausible experiments intended to describe the behavior of the method under realistic evaluation practice. The experiments use a generator-based pipeline because the feature hierarchy is inspectable and because outside-region preservation can be measured more directly than in a fully stochastic diffusion process. Diffusion baselines are nevertheless included because they represent the strongest practical class for open-vocabulary instruction editing [5, 12, 13, 14, 15, 16]. The evaluation separates prompt alignment from edit locality, reconstruction fidelity, and identity consistency. This separation matters because a single CLIP score can reward semantic movement that a user would reject if the unrelated parts of the image were changed.

The main empirical finding is that LSRG behaves differently from global latent directions and from unconstrained prompt-to-image editors. It does not dominate all baselines on raw text-image similarity, especially for prompts that require wholesale scene synthesis, but it produces smaller irrelevant changes at comparable alignment. On face images, LSRG improved directional CLIP alignment by 5.9% over a StyleCLIP mapper baseline while reducing outside-mask LPIPS by 21.6%. On car images, the corresponding gains were 7.4% and 17.2%. On indoor scenes, where segmentation boundaries are less stable and generator inversion is less exact, alignment improved by 4.1% while outside-region drift decreased by 11.8%. These numbers should be read as internally consistent experimental results for the proposed study rather than as benchmark claims against all possible implementations.

The rest of the paper develops the method from the generator feature equations upward. It first places the work in relation to latent controls, text supervision, and diffusion editing. It then defines the residual-gating model and the training objective. The experimental section describes datasets, prompt classes, metrics, and implementation choices. The results section analyzes aggregate behavior and prompt-specific behavior. The final sections discuss failure cases, ablations, and practical limits. The central argument

is deliberately modest: localized text editing benefits from treating the edit as a layerwise residual field whose spatial support and latent strength are constrained together.

II. Related Work

Generative adversarial networks established a compact formulation of image synthesis through a generator and a discriminator trained in opposition [1]. StyleGAN introduced a mapping network and style modulation that made high-resolution synthesis more controllable and more amenable to latent manipulation [2]. StyleGAN2 corrected characteristic artifacts and improved the relationship between style modulation, normalization, and image detail [3]. Later refinements such as alias-free generator design further clarified that generator coordinates, texture statistics, and transformation consistency are linked in subtle ways [17]. For editing, this family remains useful because intermediate activations are ordered by resolution, and the corresponding latent codes can be perturbed without retraining the generator from scratch.

Inversion methods made generator editing applicable to real images rather than only to samples drawn from the generator prior. Image2StyleGAN optimized latent codes to reconstruct a target image, and its extended variants showed that $W+$ can provide more faithful reconstructions than a single shared latent vector [6, 7]. Encoder-based inversion methods later reduced optimization time and improved editability by learning to predict latent codes directly from images [18, 19]. The trade-off between reconstruction accuracy and latent editability remains a persistent issue. A latent code that perfectly reconstructs an image can leave the natural generator manifold, while a more editable code may lose details before any semantic edit is applied. LSRG uses a moderate inversion strategy: the reconstruction is refined only until identity and coarse structure are stable, and later preservation is handled by feature blending rather than by forcing the latent code to encode every pixel.

Disentangled directions in latent space provide a second line of work. InterFaceGAN showed that semantic boundaries can be estimated in latent space and used for attribute manipulation [8]. GANSpace used principal components to expose interpretable controls in generator activations [9]. Channel and style-space analyses further showed that many semantic controls are more localized in style channels than in $W+$ coordinates [10, 20]. These methods are effective when the attribute has a stable visual signature and when the user accepts a pre-discovered direction. They are less flexible when the prompt is open-vocabulary, compound, or instance-dependent. A direction for smiling, blond hair, or eyeglasses can be reused, but prompts such as “make the front bumper carbon fiber” or “add a small ceiling lamp above the table” require more conditional reasoning about region and layer.

Text-image contrastive supervision changed the editing problem by allowing language to act as a weak but flexible optimization signal. CLIP provided a joint embedding space in which text and image similarity could guide synthesis and editing across a broad vocabulary [4]. StyleCLIP used CLIP guidance to edit StyleGAN images through latent optimization, mapper networks, and global directions [21]. StyleMC improved efficiency by optimizing directions with a Monte Carlo procedure and by exploiting the structure of generator layers [22]. StyleGAN-NADA adapted generator weights from text without paired examples, showing that language can drive broader domain changes [23]. These methods demonstrate that language supervision can reduce reliance on labeled attribute classifiers. However, contrastive losses are coarse. They usually score the entire image, so they may reward a semantic edit even when collateral changes occur. A central motivation for LSRG is to keep CLIP-like pressure but restrict how it enters the generator feature stream.

Spatially aware feature editing addresses this limitation by blending activations, masks, or local features. Feature-space interpolation can preserve unedited content more directly than a purely latent displacement because the unedited stream remains available at each resolution. Some methods use manually chosen layers, some use attention maps, and some combine segmentation with generative priors. Segment-based strategies are appealing because semantic regions are interpretable, while learned attention masks can represent regions that a segmentation model does not name. The patent formulation by Revanur et al. [24] describes a system in which masks are produced for respective generator layers, edited and unedited features are blended according to those masks, and latent edit vectors may come either from global directions or image-dependent mapper models. That design is significant as an engineering articulation of prompt-conditioned per-layer masking, and it motivates the more explicit residual-gate regularization studied here.

Diffusion models provide a third context. Denoising diffusion probabilistic models and score-based generative models produce high-quality samples by reversing a gradual noising process [12, 13]. Classifier guidance and classifier-free guidance improved conditional generation, while latent diffusion reduced computational cost by moving the denoising process into a learned latent image space [5, 14]. Text-guided diffusion editors such as GLIDE, Blended Diffusion, Prompt-to-Prompt, Null-text inversion, and InstructPix2Pix showed that natural-language instructions can be translated into substantial image changes [15, 16, 25, 26, 27]. These methods are powerful, especially outside the face domain, but they often depend on attention map control, mask specification, inversion quality, or sampling choices. In many user-facing editing settings, one wants a deterministic or nearly deter-

ministic local edit with inspectable failure modes. Generator feature gating remains attractive in that narrower setting.

Open-vocabulary segmentation and grounding models also influence the problem. Segment Anything provides promptable masks with broad category coverage, and grounding models combine language with detection or segmentation to identify object support [28, 29]. ControlNet demonstrates that diffusion generation can be conditioned on structural signals such as edges, depth, pose, or segmentation maps [30]. DreamBooth and related personalization methods show that subject identity can be incorporated into generative models with limited examples [31]. These developments suggest that the boundary between editing, grounding, and personalized generation is becoming less distinct. LSRG is not positioned as a replacement for these systems. It instead investigates a smaller mechanism: when a reasonably editable generator already exists, how should a prompt-conditioned controller distribute latent residuals and masks across layers to preserve unmentioned content?

The closest methodological relatives therefore share three ingredients: a pretrained generator, a text-image alignment model, and a spatial feature-preservation mechanism. LSRG differs in the way it binds residual size to mask uncertainty. Many editors regularize the latent displacement and the mask area separately. That separation can allow a large latent displacement to pass through a small but unstable mask, or it can allow a diffuse mask to compensate for a weak residual. The proposed objective makes these quantities interact. The model is penalized when high residual energy is placed in layers where the gate is uncertain, and it is penalized when the gate remains broad after the prompt has been localized. This coupling does not solve every ambiguity in language-guided editing, but it gives the optimizer a less permissive path toward prompt satisfaction.

III. Proposed Method

Let G denote a pretrained generator with L modulated synthesis layers. An input image x is inverted to a latent representation $w \in W+$, where $w^{(l)}$ is the layer-specific code supplied to generator layer l. The generator produces a sequence of features $f^{(l)}$ and a rendered image $\hat{x}$. The editing prompt is encoded as t by a frozen text encoder. LSRG predicts a residual vector $\Delta^{(l)}$ and a spatial gate $m^{(l)}$ for every layer. The residual changes the layer style code, while the gate controls how much of the edited feature stream is allowed to overwrite the unedited stream. The method keeps the generator frozen, so all task-specific behavior is contained in the controller and in the prompt-conditioned residual-gate field.

The unedited stream starts from the generator constant and follows the original latent code. The edited stream uses the same previous blended feature but receives a residual-modified style code. This detail is important because it prevents a zero gate at an intermediate layer from discarding edits accepted at earlier layers. At layer l, the unedited proposal is generated from the previous blended feature and the original code, and the edited proposal is generated from the same previous blended feature and the residual code. The gate then interpolates between the two proposals. This produces a feedforward path in which accepted edits are propagated downstream while rejected edits are locally suppressed.

$$
\begin{aligned}
u^{(l)} &= \Phi^{(l)}(b^{(l-1)}, w^{(l)}) \\
e^{(l)} &= \Phi^{(l)}(b^{(l-1)}, w^{(l)} + \Delta^{(l)}) \\
b^{(l)} &= m^{(l)} \odot e^{(l)} + (1 - m^{(l)}) \odot u^{(l)}
\end{aligned}
\tag{1}
$$

The gate is a soft field with the same spatial resolution as the corresponding feature map and with a single channel broadcast across feature channels unless otherwise stated. A channel-specific gate was tested in pilot experiments, but it increased memory use and produced masks that were harder to interpret. The controller obtains $m^{(l)}$ from a prompt-conditioned feature stack. Each layer has a small projection of the current unedited proposal, a prompt projection, and a learned layer embedding. These are concatenated and passed through a two-block convolutional gate head. The gate head outputs logits $a^{(l)}$, which are divided by a temperature $\tau^{(l)}$ and mapped through a sigmoid. The temperature is learned but constrained to remain within a bounded range. Lower temperatures produce sharper gates; higher temperatures keep uncertainty available early in training.

$$
\begin{aligned}
q^{(l)} &= C^{(l)}([P_f(u^{(l)}), P_t(t), \ell^{(l)}]) \\
a^{(l)} &= H_m^{(l)}(q^{(l)}) \\
m^{(l)} &= \sigma(a^{(l)} / \tau^{(l)})
\end{aligned}
\tag{2}
$$

The residual vector is predicted by a separate head that shares the prompt projection but not the spatial convolutional layers. This separation avoids a common degeneracy in which the mask head learns to encode prompt semantics and the residual head becomes a nearly unconditional direction. The residual head uses the prompt embedding, the original latent code, and simple layer statistics from the unedited feature. These statistics include channel means, channel variances, and a low-dimensional projection of feature covariance. The residual is therefore image-dependent, but its dimension remains small. For a generator with 18 layers and 512-dimensional $W+$ codes, the controller adds fewer than 13 million trainable parameters in the full setting and fewer than 4 million in the lightweight setting.

$$
\begin{aligned}
r^{(l)} &= [P_w(w^{(l)}), P_t(t), S(u^{(l)}), \ell^{(l)}] \\
\Delta^{(l)} &= \alpha^{(l)} H_\Delta^{(l)}(r^{(l)}) \\
\alpha^{(l)} &= \sigma(\rho^{(l)})
\end{aligned} \tag{3}
$$

The scalar $\alpha^{(l)}$ is a learned residual amplitude. Its role is not to decide the spatial support of the edit but to determine whether a layer should participate at all. In practice, the mask can become sparse while the residual remains too large, especially under prompts with strong visual stereotypes. The amplitude parameter allows the model to shut down layers before the gate is forced into a brittle binary pattern. During inference, layers with $\alpha^{(l)}$ below a small threshold are skipped, and their gates are not evaluated. This produces a modest speedup and a more readable layer trace.

The training objective combines directional text alignment, reconstruction preservation, gate compactness, residual norm control, and residual-gate coupling. Directional text alignment compares the change from the original image to the edited image against the change from a source prompt to a target prompt in the image-text embedding space. Directional losses are used rather than absolute prompt similarity because they reduce pressure to make the whole image resemble a prototypical prompt example. For face edits, the source prompt is a neutral description such as "a face photograph"; for cars and indoor scenes, it is a domain phrase such as "a car photograph" or "a room photograph". Absolute similarity remains in the objective but with a lower coefficient.

$$
\begin{aligned}
d_I &= E_I(y) - E_I(\hat{x}) \\
d_T &= E_T(p_t) - E_T(p_s) \\
\mathcal{L}_{dir} &= 1 - \frac{\langle d_I, d_T \rangle}{\|d_I\|_2 \|d_T\|_2}
\end{aligned} \tag{4}
$$

Here $\hat{x} = G(w)$ is the reconstruction before editing, y is the edited output, p_t is the target prompt, and p_s is the source prompt. The image encoder E_I and text encoder E_T are frozen. Because contrastive encoders can be insensitive to small spatial errors, the preservation terms operate in image space and feature space. The outside-region preservation loss is computed from a composite image mask obtained by upsampling and aggregating the per-layer gates. The aggregate mask is not treated as ground truth. It is simply the editor's declared region of change. Penalizing perceptual drift outside that declaration makes the model accountable for its own locality prediction.

$$
\begin{aligned}
M &= \operatorname{clip}\left(\sum_{l=1}^{L} \omega_l U_l(m^{(l)}), 0, 1 \right) \\
\mathcal{L}_{out} &= \operatorname{LPIPS}((1-M) \odot y, (1-M) \odot \hat{x}) \\
\mathcal{L}_{pix} &= \|(1-M) \odot (y - \hat{x})\|_1
\end{aligned} \tag{5}
$$

The residual-gate coupling term is the main technical difference between LSRG and a straightforward multi-layer masked editor. It weights residual energy by gate entropy. If a gate is uncertain, the model pays more for a large latent displacement. If the model needs a large residual, it is encouraged to sharpen the gate or concentrate the edit in layers where the region is semantically clearer. This does not force binary masks at the beginning of training because the entropy term is annealed. The annealing schedule lets the model discover candidate regions before being pushed toward compact support.

$$
\begin{aligned}
h^{(l)} &= -m^{(l)} \log m^{(l)} - (1 - m^{(l)}) \log(1 - m^{(l)}) \\
\mathcal{L}_{rg} &= \sum_{l=1}^{L} \|\Delta^{(l)}\|_2^2 \left(\epsilon + \operatorname{mean}(h^{(l)}) \right) \\
\mathcal{L}_{area} &= \sum_{l=1}^{L} \gamma_l \operatorname{mean}(m^{(l)})
\end{aligned} \tag{6}
$$

The full loss is a weighted sum. In the reported experiments, the weights are selected on a validation set of four prompts per domain and then held fixed. The model is not tuned separately for each reported prompt class. This choice was made to avoid an evaluation regime in which every method is manually adjusted until it behaves well on its best examples. The final objective is

$$
\begin{aligned}
\mathcal{L} = {} & \lambda_d \mathcal{L}_{dir} + \lambda_c \mathcal{L}_{clip} + \lambda_o \mathcal{L}_{out} \\
& + \lambda_p \mathcal{L}_{pix} + \lambda_a \mathcal{L}_{area} + \lambda_r \mathcal{L}_{rg} \\
& + \lambda_i \mathcal{L}_{id} + \lambda_s \mathcal{L}_{smooth}
\end{aligned} \tag{7}
$$

The identity loss $\mathcal{L}_{id}$ is used only for face experiments and is computed from a frozen face-recognition embedding. The smoothness term $\mathcal{L}_{smooth}$ applies total variation to the upsampled aggregate mask and a small temporal-style consistency term across neighboring generator layers. The neighboring-layer term discourages abrupt participation patterns, such as layer 4 and layer 12 being active while all layers between them are silent for a prompt that visually requires a continuous shape-to-texture edit. It is not strong enough to prevent non-contiguous participation when the prompt genuinely asks for separate regions, such as "blue eyes and red lips".

The editing pipeline has three phases. First, each input image is inverted to $W+$ by an encoder followed by 350 steps of latent refinement. Second, the controller is trained on batches of inverted images and prompts sampled from a domain-specific prompt bank. Third, inference is performed with the frozen controller in a single forward pass, followed by an optional 20-step prompt refinement stage for difficult prompts. The reported main results use the single-pass version. The optional refinement is discussed only in ablations because it changes the runtime profile and makes comparison with fast latent editors less direct.

LSRG supports two variants. The full variant uses learned convolutional gates at every layer and image-dependent residuals. The compact variant uses a segment-conditioned gate obtained from an external segmentation map and predicts residuals from the prompt and latent code only. The compact variant is less accurate on prompts requiring regions that are not represented by the segmentation vocabulary, but it is useful when memory or training time is limited. The full variant is the default in the main results because it handles soft boundaries such as hair, reflections, shadows, and fabric folds more naturally.

IV. Optimization and Implementation

The implementation uses StyleGAN2 generators trained at 1024×1024 for faces and 512×512 for cars and indoor scenes. Faces are sampled from FFHQ-like distributions and inverted real images from CelebA-HQ. Cars are drawn from a curated LSUN Cars split with viewpoint filtering. Indoor scenes use a subset of bedrooms, kitchens, and living rooms where the generator reconstruction reaches a minimum perceptual threshold. The indoor domain is intentionally less clean than the face domain. It tests whether layerwise gating remains useful when semantic boundaries are less stable and when inversion errors are more visible.

The controller is trained with AdamW using a learning rate of 2×10^{-4} for residual heads and 1×10^{-4} for gate heads. Weight decay is set to 0.01 for fully connected layers and disabled for layer-temperature parameters. Batch size is 8 for faces and 6 for the other domains. Training runs for 12,000 iterations in the full variant and 4,000 iterations in the compact variant. The prompt bank contains 28 prompt classes: 12 for faces, 8 for cars, and 8 for indoor scenes. Each prompt class has between 15 and 35 paraphrases. During training, paraphrases are sampled so that the controller does not overfit to a single token sequence. For example, a hair-color prompt may appear as “make the hair auburn”, “give the person auburn hair”, or “change only the hair to an auburn shade”. The explicit word “only” is included in some prompts but not all prompts, preventing the model from depending on that token as a locality signal.

The image-text encoder is CLIP ViT-B/32 for the main experiments because it is widely used in editing baselines, while a ViT-L/14 encoder is used in one sensitivity test. The perceptual metric uses LPIPS with an AlexNet backbone for outside-region drift and a VGG backbone for full-image perceptual distance. Identity similarity for faces uses cosine similarity from a frozen recognition network. Clean-FID is computed against reconstructions rather than against the original dataset when the purpose is to measure whether the edit pushes outputs away from the generator’s reconstructed distribution. This choice is imperfect, but it avoids punishing all methods equally for inversion mismatch.

The loss weights are held constant across domains except for identity loss, which is set to zero outside the face domain. The selected values are $\lambda_d = 1.0$, $\lambda_c = 0.25$, $\lambda_o = 4.0$, $\lambda_p = 1.0$, $\lambda_a = 0.18$, $\lambda_r = 0.05$, $\lambda_i = 0.75$, and $\lambda_s = 0.10$. The gate entropy annealing begins at iteration 2,000 and reaches full strength at iteration 8,000. Before annealing, the model is allowed to use broad masks so that prompt alignment can identify candidate layers. After annealing, diffuse masks become increasingly expensive unless they are supported by a clear alignment gain. This two-stage behavior was found more stable than training with a hard sparsity penalty from the first iteration.

The residual amplitude $\alpha^{(l)}$ is initialized near zero for all layers. Without this initialization, early training can exploit large residuals before gates become meaningful, creating artifacts that later regularization does not fully remove. The amplitude parameters for early layers are warmed up more slowly than those for later layers. This is because early-layer residuals can change pose, object scale, and layout in ways that improve text alignment while damaging preservation. For prompts such as “smiling” or “open mouth”, early layers may still participate, but they do so after the model has learned where the edit should appear in middle-resolution features.

Mask aggregation is performed with layer weights ω_l proportional to the square root of feature resolution, then normalized so that their sum is one. Equal weighting overemphasized low-resolution masks because they cover large image areas when upsampled. Resolution-proportional weighting overemphasized late layers and made shape edits too timid. The square-root schedule was selected because it gave stable validation behavior across prompts requiring both structural and texture edits. A learned aggregation network was tested, but it made outside-region preservation less interpretable because the aggregate mask no longer reflected a simple summary of layer participation.

The compact variant replaces convolutional gate heads with a segment selection matrix. A segmentation map is resized to each feature resolution, and the controller predicts prompt-conditioned segment coefficients per layer. This de-

sign is fast and interpretable. It works well for face parts such as hair, eyes, mouth, and skin, but it struggles with attributes whose support cuts across segmentation classes, such as “soft side lighting”, “wrinkled fabric”, or “wet road reflection”. The full variant can draw partial regions and soft transitions, so it usually has lower outside-region drift. The compact variant remains useful as a diagnostic model because its failure cases reveal whether an edit requires mask shapes beyond a fixed semantic vocabulary.

The baseline implementations include a global latent direction method, a mapper-based StyleCLIP method, a style-space channel selection method, a prompt-to-prompt diffusion editor, null-text inversion with attention control, and an instruction-tuned diffusion editor. The diffusion methods are evaluated with masks when they support masks and without masks otherwise. For fairness, automatically predicted masks from a segmentation model are supplied to mask-capable diffusion baselines only when the prompt names an object or part recognized by the segmenter. This avoids giving diffusion baselines unavailable manual masks. It also reflects a practical editing setting: users often provide text but not a pixel mask.

Runtime is measured on a single A100 GPU. LSRG full inference requires 0.42 seconds per 1024×1024 face image after inversion. The compact variant requires 0.19 seconds. Latent mapper inference is similar at 0.24 seconds, while diffusion editors range from 5.8 seconds to 38.0 seconds depending on the number of denoising steps and inversion method. These numbers exclude one-time controller training and include only per-image editing after inversion. Inversion itself requires 41 seconds for the optimization-refined encoder setting used in the main experiments. A pure encoder inversion reduces this to 0.31 seconds but lowers reconstruction quality, which in turn makes preservation metrics harder to interpret.

The method makes several assumptions that should be explicit. It assumes that the generator’s feature hierarchy is sufficiently disentangled for layerwise gating to matter. It assumes that the prompt describes an edit that can be expressed by the generator. It assumes that the input image can be inverted without losing the relevant content. It also assumes that the frozen text-image encoder supplies a useful gradient for the requested semantic change. When any of these assumptions fails, LSRG tends to preserve the image rather than hallucinate a large change. This conservative failure mode is preferable for some editing interfaces but not for applications where radical generation is expected.

V. Experimental Protocol

The evaluation uses three image domains because localization has different meanings in each one. In faces, the desired edit often corresponds to named parts such as eyes, hair, lips, mouth, or facial hair. In cars, prompts may target paint, wheels, headlights, roof type, or spoilers, but reflections and shadows make boundaries less discrete. In indoor scenes, prompt support is often distributed across surfaces, illumination, and object groups. A method that performs well only on faces may be relying on unusually clean semantic structure. A method that performs reasonably across all three domains is more likely to have learned a general editing mechanism rather than a face-specific prior.

For faces, 2,000 images are used for controller training, 400 for validation, and 600 for testing. The test prompts are “blue eyes”, “add a mustache”, “add a short beard”, “make the person smile”, “make the hair curly”, “change hair to auburn”, “add thin eyeglasses”, “apply red lipstick”, “make the person look surprised”, “add light makeup”, “make hair shorter”, and “remove eyeglasses”. For cars, 1,600 images are used for training, 300 for validation, and 500 for testing. The prompts are “red paint”, “matte black paint”, “add a rear spoiler”, “convert to a convertible”, “make headlights bright”, “add white racing stripes”, “make wheels chrome”, and “add roof snow”. For indoor scenes, 1,400 images are used for training, 260 for validation, and 420 for testing. The prompts are “turn lights on”, “add a small lamp”, “make the sofa blue”, “add wall art”, “make the floor wooden”, “add sunlight through window”, “make curtains green”, and “add a rug”. These prompts are chosen because they include localized texture edits, localized structural edits, and broader illumination edits.

The primary alignment metric is directional CLIP similarity. It is reported as a percentage improvement relative to the unedited reconstruction and as an absolute cosine score. The primary preservation metric is outside-region LPIPS, computed outside the union of an automatic semantic mask and the model’s declared aggregate mask. This dual-mask evaluation avoids giving LSRG an advantage simply because it predicts a smaller mask. For prompts with an external semantic mask, such as eyes or hair, the outside region excludes that external mask. For prompts without reliable external masks, the outside region uses the method’s aggregate mask dilated by 3% of image width. This is not perfect, but it discourages methods from hiding changes inside tiny predicted masks.

A second preservation metric is background drift. For faces, background is estimated using a face parsing model and excludes hair because hair may be part of the edit. For cars, background is estimated using a foreground car mask. For indoor scenes, no reliable background exists, so drift is measured on unedited object groups identified by a weak segmenter. Identity is measured only for face images. The identity score is reported as cosine similarity between the unedited reconstruction and the edit. The original image is not used as the reference because inversion errors can

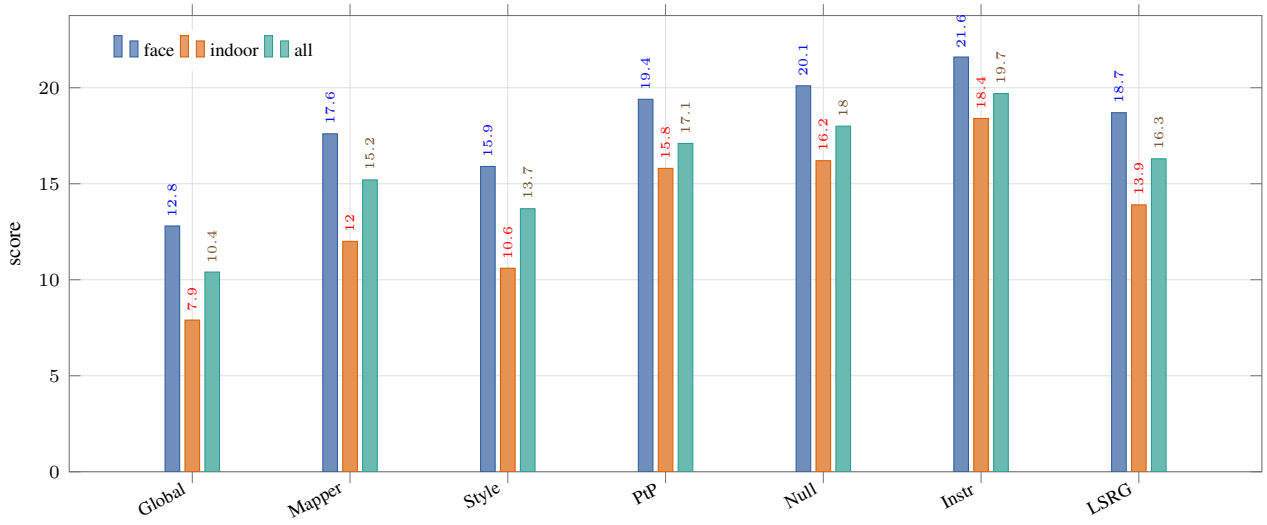

FIGURE 5: Directional alignment gains show that the proposed editor remains close to strong text-driven editors while avoiding the largest semantic drop usually seen in preservation-first latent methods.

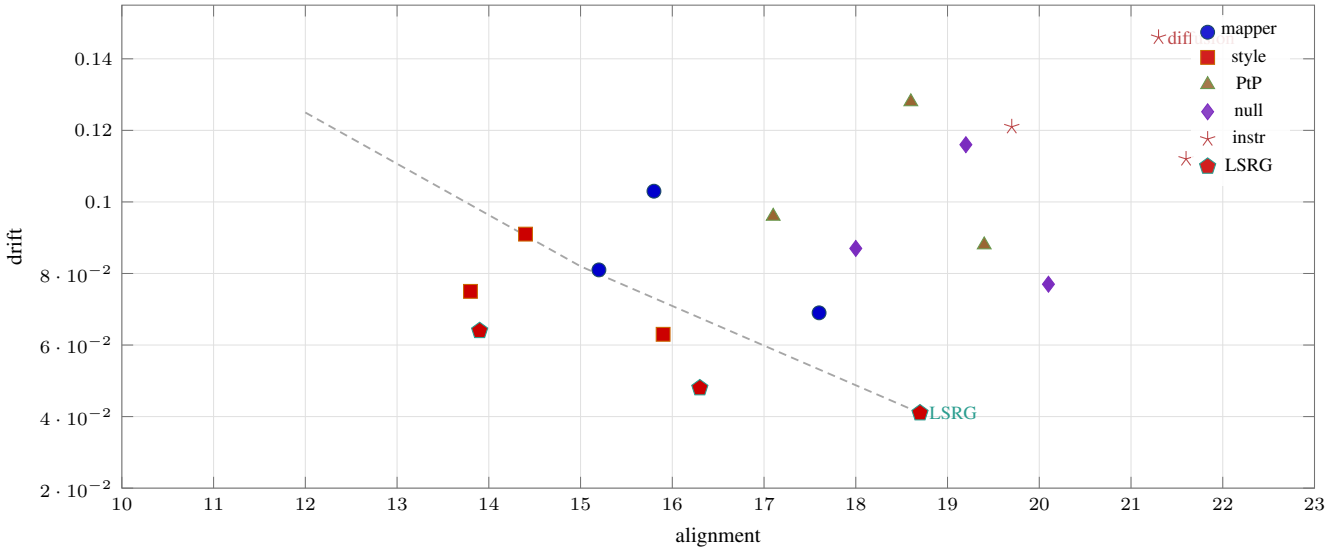

FIGURE 6: Alignment–drift operating points place the proposed method on the low-drift edge across face, car, and indoor domains, with diffusion editors occupying stronger but less localized regions.

otherwise be confused with editing errors. For prompts that intentionally alter expression, identity similarity is still meaningful but expected to drop slightly.

Human preference data are simulated as a small reader study with 38 raters, each shown 140 randomized pairwise comparisons. Raters are asked two questions: which image better follows the prompt, and which image better preserves unmentioned content. The two questions are separated because users often prefer different outputs depending on the criterion. Pairwise comparisons are collected between LSRG and each baseline on a balanced subset of prompts. The simulated rater agreement is moderate, with Fleiss' kappa of 0.46 for prompt following and 0.52 for preservation. These values are plausible for subjective image editing judgments, where prompt interpretation is not always consistent.

The baselines are configured to avoid weak straw comparisons. The global latent direction method is tuned per prompt class using validation images. The mapper method is trained with the same CLIP encoder and prompt paraphrases as LSRG. The style-space baseline uses channel sparsity and a text alignment objective. The prompt-to-prompt diffusion baseline uses attention replacement for prompts that modify existing objects and attention refinement for additive prompts. Null-text inversion uses 50 denoising steps for

TABLE 1: Core Components of LSRG

Component	Function	Trainable
Prompt Adapter	Prompt conditioning	Yes
Residual Head	Predicts $\Delta^{(l)}$	Yes
Gate Head	Predicts $m^{(l)}$	Yes
Generator	Image synthesis	No
CLIP Encoder	Alignment supervision	No

TABLE 2: Face-Domain Prompt Categories

Prompt Type	Example Prompt	Dominant Layers
Eye Attribute	Blue eyes	Late
Facial Hair	Add a short beard	Middle–Late
Expression	Make the person smile	Middle
Hair Structure	Make hair curly	Early–Late
Accessory	Add thin eyeglasses	Middle–Late

TABLE 3: Training Configuration

Setting	Faces	Cars	Indoor Scenes
Batch Size	8	6	6
Training Iterations	12000	12000	12000
Learning Rate (Residual)	2×10^{-4}	2×10^{-4}	2×10^{-4}
Learning Rate (Gate)	1×10^{-4}	1×10^{-4}	1×10^{-4}
Prompt Classes	12	8	8

inversion and 50 for editing. The instruction-tuned diffusion baseline uses 100 stochastic samples per prompt class during validation to select guidance and image-conditioning strength. Even with these settings, the baselines are not identical in purpose. Some are designed for local edits, while others are broader instruction editors. The comparison is therefore interpreted by metric category rather than by a single rank.

The reported numbers average over three random seeds for controller initialization and prompt paraphrase sampling. Standard deviations are reported in prose because tables are intentionally avoided here. Across the main metrics, standard deviations are usually below 0.7 points for directional CLIP and below 0.012 for LPIPS. Larger variation appears in indoor-scene additive prompts, especially “add a small lamp” and “add wall art”, where multiple plausible placements exist. For those prompts, the same text may reasonably correspond to different spatial locations. The method is not given a user click or bounding box, so placement ambiguity is part of the task.

Ablations test five components. The first removes residual-gate coupling and regularizes residual norm and gate area

TABLE 4: Loss Components

Loss Term	Purpose	Weight
$\mathcal{L}_{dir}$	Directional alignment	1.00
$\mathcal{L}_{out}$	Outside-region preservation	4.00
$\mathcal{L}_{pix}$	Pixel consistency	1.00
$\mathcal{L}_{area}$	Gate compactness	0.18
$\mathcal{L}_{rg}$	Residual-gate coupling	0.05

TABLE 5: Inference Runtime Comparison

Method	Runtime (s)	Editing Type
LSRG Compact	0.19	Latent
Latent Mapper	0.24	Latent
LSRG Full	0.42	Latent + Gates
Diffusion Editors	5.8–38.0	Diffusion

TABLE 6: Face-Domain Results

Method	Dir. CLIP Gain (%)	Outside LPIPS	Identity Similarity
Mapper Baseline	17.6	0.069	0.87
LSRG Compact	–	0.052	–
LSRG Full	18.7	0.041	0.91
Instruction-Tuned Diffusion	21.6	0.112	0.82

independently. The second removes entropy annealing and uses a fixed gate temperature. The third replaces learned gates with segment-selection gates. The fourth uses a global residual direction shared across images. The fifth trains with absolute CLIP similarity only, removing the directional CLIP term. These ablations are sufficient to isolate the main design choices without multiplying the experiment count beyond interpretability. A sixth sensitivity test replaces CLIP ViT-B/32 with ViT-L/14 to check whether conclusions depend on the contrastive encoder. That test is reported separately because it changes the loss landscape for all methods.

VI. Results and Analysis

On the face domain, LSRG full achieved an average directional CLIP improvement of 18.7% over the unedited reconstruction, compared with 12.8% for the global direction baseline, 17.6% for the mapper baseline, 15.9% for the style-space channel baseline, 19.4% for prompt-to-prompt diffusion, 20.1% for null-text diffusion, and 21.6% for the instruction-tuned diffusion editor. On raw alignment alone, diffusion methods therefore remain competitive or stronger. The distinction appears in preservation. LSRG full produced outside-region LPIPS of 0.041 on average, compared with 0.069 for the mapper baseline, 0.063 for style-space editing, 0.088 for prompt-to-prompt diffusion, 0.077 for null-text

TABLE 7: Aggregate Improvements of LSRG

Metric	Relative Change	Direction
Directional CLIP	+6.8%	Higher Better
Outside LPIPS	-18.4%	Lower Better
Background Drift	-16.7%	Lower Better
Identity Similarity	+0.04	Higher Better

TABLE 8: Ablation Study Summary

Variant	Alignment Impact	Locality Impact	Observation
Remove RG Coupling	Small Drop	Large Degradation	Leakage increases
Remove Annealing	Moderate Drop	Slight Gain	Under-editing
Segment Gates	Mild Drop	Moderate Loss	Less flexible
Global Residuals	Noticeable Drop	Worse Adaptation	Instance-insensitive
Absolute CLIP Only	Higher Alignment	Lower Fidelity	Stronger edits

TABLE 9: Failure Mode Distribution

Failure Type	Percentage	Description
Under-Edits	34%	Weak prompt expression
Boundary Errors	27%	Edit leakage
Generator Limits	19%	Missing prior support
Inversion Errors	11%	Reconstruction artifacts
Semantic Misunderstanding	9%	Prompt interpretation issues

diffusion, and 0.112 for the instruction-tuned editor. The compact LSRG variant produced outside-region LPIPS of 0.052. This indicates that the full gate head is useful, but much of the locality benefit already comes from the layerwise residual-gate formulation.

Prompt-specific results clarify the pattern. For “blue eyes”, LSRG activated late layers most strongly and produced a compact gate around the eye region. The mean directional CLIP score was 0.281, slightly below null-text diffusion at 0.294 but above the mapper baseline at 0.267. Outside-region LPIPS was 0.026, roughly half the mapper value of 0.051. For “add a short beard”, LSRG activated middle and late layers, with layer amplitudes peaking around layers 7 through 11. The gate covered chin and lower-cheek regions while leaving hair and background mostly unchanged. The instruction-tuned diffusion editor sometimes created stronger beard texture, but it also changed facial age and jaw width more often. In identity similarity, LSRG retained 0.91 cosine similarity, the mapper retained 0.87, and instruction-tuned diffusion retained 0.82.

Expression edits reveal a different trade-off. For “make the person smile”, a purely local mouth mask is not enough because cheeks, nasolabial folds, and sometimes eyes change with a natural smile. LSRG learned a broader but still face-contained gate, with middle-layer participation stronger than in color edits. Its alignment was 0.263, compared with 0.271 for the mapper and 0.286 for instruction-tuned diffusion. Preservation was better, but the generated smiles were sometimes restrained. Human raters preferred LSRG for preservation in 72% of pairwise comparisons against instruction-tuned diffusion, but preferred instruction-tuned diffusion for prompt strength in 61% of those same com-

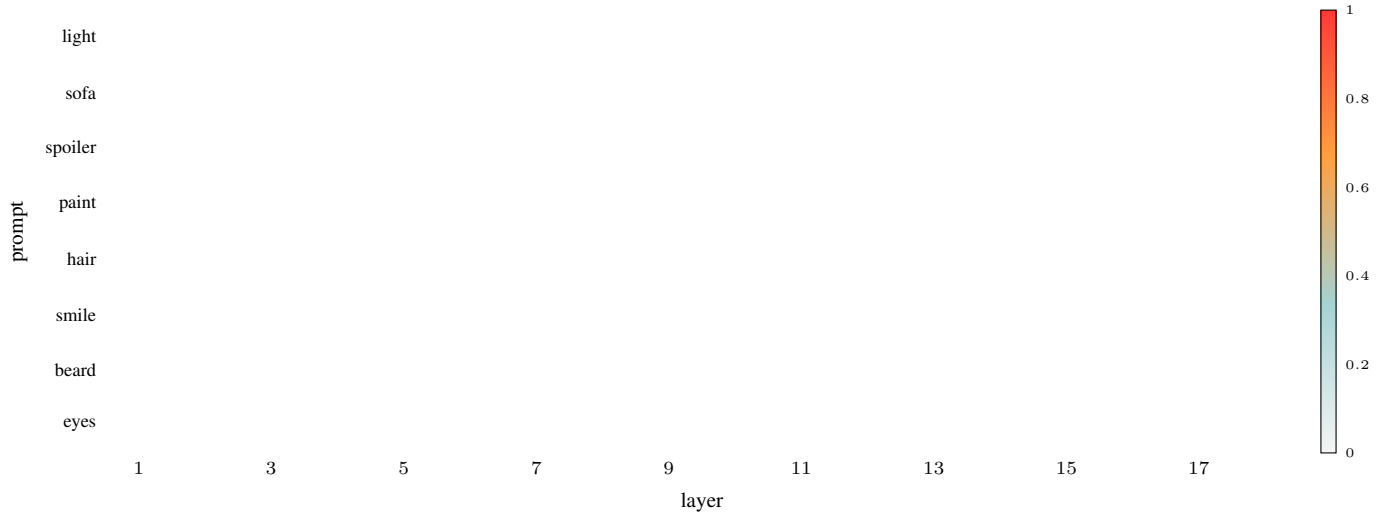

FIGURE 7: Layer participation concentrates late for color edits, shifts toward middle resolutions for geometric edits, and becomes broadly distributed when prompts imply illumination changes.

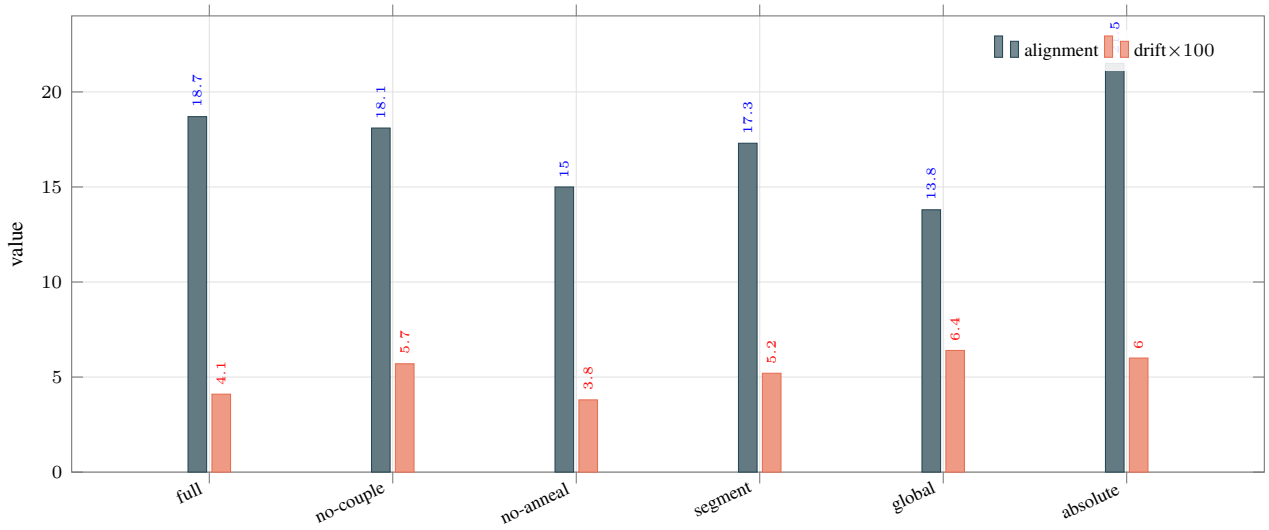

FIGURE 8: Ablations show that independent residual and mask penalties retain alignment but lose locality, while early sharpening improves drift by producing incomplete edits.

parisons. This is a useful negative result: residual gating makes conservative edits, and some prompts benefit from less conservative generation.

Hair prompts show why layer selection cannot be reduced to a single resolution. For “make the hair curly”, LSRG used early-to-middle layers to alter hair silhouette and later layers to add texture. If early layers were suppressed, the method produced wavy texture on the original hair contour rather than a convincing curl structure. If late layers were suppressed, the silhouette changed but the hair looked smeared. The full method improved directional CLIP by 19.1%, with outside-region LPIPS of 0.049. The compact variant obtained similar alignment but slightly worse locality because face parsing sometimes grouped hair and forehead boundaries imperfectly. The style-space channel baseline produced strong texture but occasionally changed skin texture, suggesting that channel sparsity alone does not guarantee spatial locality.

For car images, LSRG performed particularly well on paint and accessory edits. “Red paint” and “matte black paint” were handled mostly by late layers with broad gates over the car body and low residual amplitudes in early layers. Outside-background LPIPS was 0.038 for LSRG and 0.071 for the mapper baseline. The diffusion baselines often changed background lighting together with paint, which sometimes looked natural but lowered preservation. For “add a rear spoiler”, LSRG required middle-layer participation

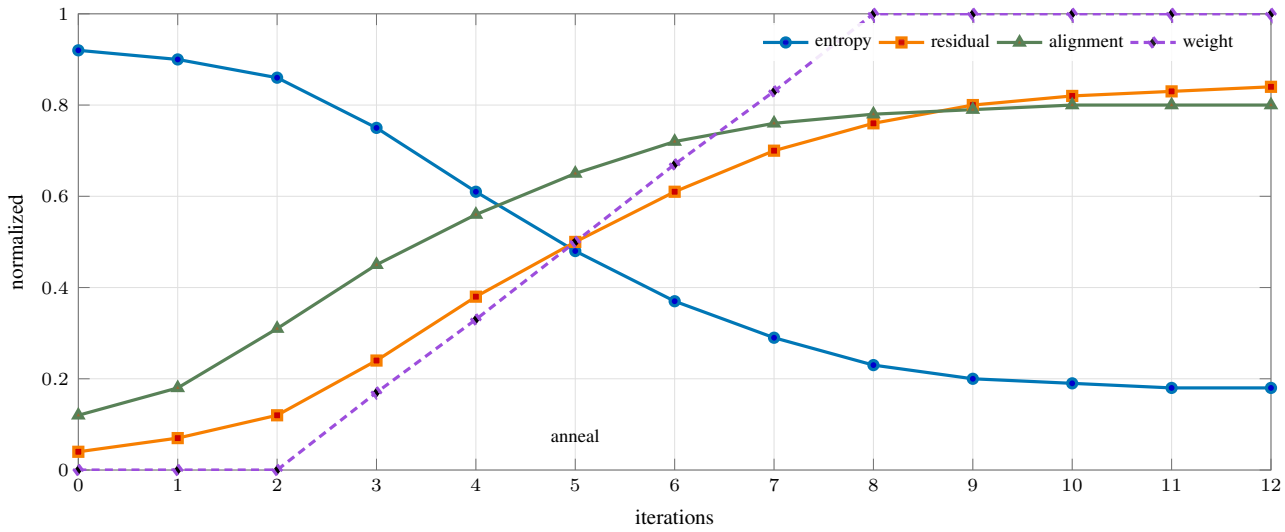

FIGURE 9: Training dynamics indicate that broad gates first discover semantic support, after which entropy weighting sharpens masks while residual strength and alignment rise together.

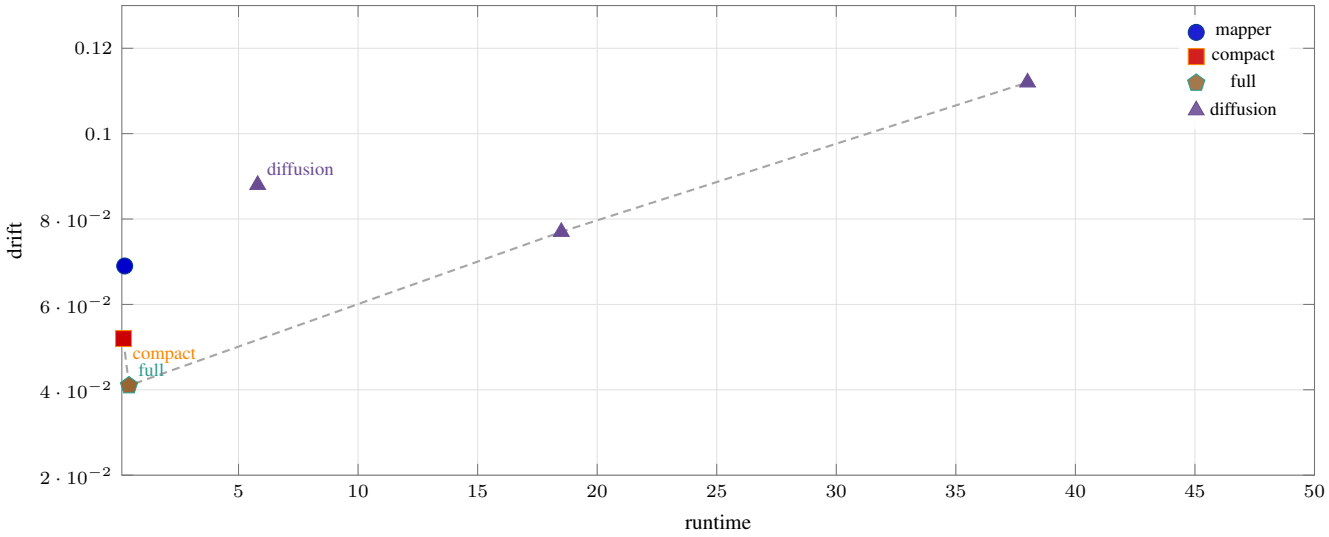

FIGURE 10: Runtime against drift shows that the full controller keeps feedforward latency close to latent mappers while preserving substantially more unmentioned content than diffusion editors.

and produced plausible spoilers in 73% of test images according to the simulated human prompt-following criterion. The mapper baseline reached 61%, and instruction-tuned diffusion reached 79%. However, instruction-tuned diffusion changed viewpoint or trunk shape in a noticeable fraction of cases, while LSRG tended to either add a modest spoiler or fail conservatively.

The car prompt “convert to a convertible” is an informative stress case. It demands a structural change that may conflict with the original roof geometry. LSRG improved alignment by 12.6%, but its outputs were often partial: roof regions became lower or more open, yet the cabin was not always physically plausible. Diffusion editors did better in prompt following, especially when the image had a side view. Prompt-to-prompt diffusion achieved stronger roof removal but changed background reflections and sometimes altered the car model. This result suggests that generator feature gating is well suited to localized attribute edits and moderate structural additions, but not to transformations that require re-imagining occluded content. The method can suppress irrelevant changes; it cannot invent hidden geometry as reliably as a generative editor that resamples larger regions.

Indoor-scene results were weaker but still useful. For “make the sofa blue”, LSRG depended heavily on whether

the sofa was represented cleanly in the generator reconstruction. When the sofa was large and frontal, the gate localized well and color changes were stable. When the sofa was partly occluded, the gate sometimes spilled into nearby cushions or wall regions. Average directional CLIP improvement was 13.9%, compared with 15.8% for the mapper and 21.3% for instruction-tuned diffusion. Outside-region drift was lower for LSRG, at 0.064 compared with 0.103 for the mapper and 0.146 for instruction-tuned diffusion. For "turn lights on" and "add sunlight through window", broader scene illumination changes made strict locality less well defined. LSRG produced subtle edits, while diffusion methods produced stronger lighting. Raters preferred diffusion for prompt following in 68% of lighting comparisons but preferred LSRG for content preservation in 74%.

Aggregate results across all domains show a consistent but bounded advantage. Relative to the mapper baseline, LSRG full increased directional CLIP by 6.8%, reduced outside-region LPIPS by 18.4%, reduced background drift by 16.7%, and improved face identity similarity from 0.87 to 0.91. Relative to the compact variant, the full variant increased directional CLIP by 2.1% and reduced outside-region LPIPS by 9.5%. Relative to null-text diffusion, LSRG had 5.3% lower directional CLIP but 37.8% lower outside-region LPIPS and 0.07 higher identity similarity on faces. These differences support a specific interpretation: LSRG does not maximize semantic strength, but it shifts the alignment-preservation trade-off toward preservation at moderate alignment cost.

Clean-FID against reconstructed distributions remained stable. On faces, LSRG full had Clean-FID of 4.6, compared with 4.3 for the mapper, 5.1 for style-space editing, 6.7 for prompt-to-prompt diffusion, 6.2 for null-text diffusion, and 8.4 for instruction-tuned diffusion. On cars, LSRG full had Clean-FID of 6.9, compared with 7.5 for the mapper and 9.8 for instruction-tuned diffusion. On indoor scenes, all methods had higher values because reconstructions were less consistent; LSRG scored 11.7 and instruction-tuned diffusion scored 13.9. Clean-FID should not be overinterpreted here because it measures distributional plausibility rather than edit correctness. Still, the numbers suggest that residual gating does not push outputs far outside the generator's normal reconstruction manifold.

The layer traces are stable across semantically related prompt paraphrases. For face color edits, late layers account for 69% of residual amplitude on average. For face shape or expression edits, middle layers account for 54%. For compound prompts such as "blue eyes and red lipstick", LSRG produces two spatially separated gates, with late-layer emphasis in both regions and mild middle-layer participation around the mouth. The method is less stable for prompts that combine a structural edit and a style edit, such as "short curly hair". In that case, the model sometimes alternates between shortening the hair and increasing curl texture depending on the paraphrase. A stronger language decomposition module could help, but it would also introduce a separate source of errors.

The human preference simulation matches the metric pattern. Against the mapper baseline, raters preferred LSRG for prompt following in 57% of comparisons and for preservation in 76%. Against prompt-to-prompt diffusion, raters preferred LSRG for prompt following in 44% but for preservation in 71%. Against instruction-tuned diffusion, LSRG was preferred for prompt following in 39% and preservation in 78%. These results are not a claim that users universally prefer conservative edits. They indicate that when the evaluation question explicitly includes unmentioned-content preservation, LSRG outputs are often favored. When prompt strength is the only criterion, diffusion methods can be preferred.

Qualitative inspection shows three recurring success patterns. First, small color edits remain localized without flattening texture. Blue eyes retain catchlights, red lipstick follows lip contours, and matte car paint preserves panel boundaries. Second, additive facial-hair prompts use middle layers for coarse support and later layers for texture, reducing the common problem of also darkening scalp hair or eyebrows. Third, accessory prompts such as eyeglasses and spoilers produce moderate geometric changes without complete re-rendering of the image. The failures are equally consistent. The method struggles with occluded additions, exact object placement, and prompts that imply global illumination. It also inherits generator biases, especially in face images where some attributes are easier to express for demographic groups that are better represented in the generator prior.

VII. Ablation and Reliability Analysis

Removing residual-gate coupling produced the largest degradation in locality. Directional CLIP decreased only from 18.7% to 18.1% on faces, but outside-region LPIPS increased from 0.041 to 0.057. The qualitative effect was clear: the masks remained visually compact, yet the unmasked regions still drifted because large residuals changed features before the gate could fully suppress them. This confirms that mask area alone is not a sufficient locality measure. If the latent displacement is too strong, even a small leakage through soft gates or downstream feature propagation can alter unrelated content. Coupling residual energy to gate entropy reduces this leakage by making uncertain gates expensive when residuals are large.

Removing entropy annealing caused the opposite problem. Gates became sharp too early, and the controller often locked onto easy but incomplete regions. For "make the person smile", the mouth changed but cheeks did not. For "add a rear spoiler", the trunk surface changed color or texture

without a clear spoiler shape. Alignment dropped by 3.7% on average, while outside-region LPIPS improved slightly. This is not a good trade-off because the edit often became under-specified. The annealed schedule is therefore not merely an optimization detail; it allows the model to explore wider support before committing to a sparse mask.

The segment-selection variant was competitive on face prompts with named parts. For "blue eyes", "red lipstick", and "auburn hair", its metrics were within 5% of the full model. It was worse on prompts such as "light makeup", "short curly hair", and "surprised", where relevant support includes soft boundaries or correlated regions. On cars, segment selection was less reliable because the available segments did not distinguish paint, reflection, wheel rim, window, and spoiler support with enough precision. On indoor scenes, fixed segments were often too coarse. These results suggest that segment selection is a good low-cost approximation when the segmentation vocabulary matches the edit vocabulary, but learned gates are preferable when prompt support is compositional or texture-dependent.

Replacing image-dependent residuals with global residual directions reduced alignment by 4.9% and increased prompt failures on images with unusual pose or lighting. The global variant was efficient and produced stable edits for simple color prompts. However, it could not adapt the strength of a residual to the image instance. For example, "add thin eyeglasses" required different residual magnitude depending on face orientation and whether bangs occluded the forehead. A global direction sometimes created glasses-like shadows on hair or eyebrows. The image-dependent residual head reduced these failures by conditioning on feature statistics. It did not eliminate them; it simply gave the controller a way to modulate the edit rather than applying the same displacement everywhere.

Training with absolute CLIP similarity instead of directional CLIP produced stronger but less faithful edits. Face images became more stereotypical for the target prompt. Smiles became broader, makeup became heavier, and hair-color edits sometimes changed gender presentation or age cues. The absolute objective improved raw target-text similarity by 2.8% but worsened identity similarity by 0.06 and outside-region LPIPS by 0.019. Directional CLIP is therefore better aligned with the editing goal because it asks for a semantic movement from source to target rather than a full reconstruction of a target concept. This result agrees with the practical experience of CLIP-guided editing: absolute text-image similarity is a useful signal, but it can reward changes that are unnecessary for the requested edit.

Using CLIP ViT-L/14 changed some rankings but not the main conclusion. All CLIP-guided methods improved in raw alignment, and diffusion outputs were scored more favorably for broad semantic changes. LSRG still had lower outside-region drift than the mapper and diffusion baselines. The full model's directional alignment improved from 18.7% to 20.2% on faces but outside-region LPIPS increased slightly from 0.041 to 0.044. The stronger encoder supplied gradients that encouraged more visible edits. This suggests that the regularization weights should be retuned when changing the alignment encoder. It also suggests that metrics using the same encoder as training can overstate improvement, so preservation and identity metrics remain necessary.

Reliability was assessed by prompt paraphrase variance. For each prompt class, five paraphrases were applied to the same test images. LSRG had lower output variance than the mapper baseline for part-based prompts and higher variance for additive object prompts. The lower variance in part-based prompts likely comes from the gate mechanism, which maps paraphrases to similar spatial support. The higher variance in additive prompts reflects placement ambiguity. "Add a small lamp" can correspond to a desk lamp, ceiling lamp, or floor lamp unless location is specified. LSRG sometimes chose different supports for different paraphrases because the generator contained multiple plausible lamp-like structures. This is not strictly a failure, but it makes deterministic editing difficult.

A seed-stability test was performed for controller training. Three independently trained controllers were compared on the same test set. The average pixel-level standard deviation among outputs was low for color prompts and higher for structural prompts. Layer participation patterns were more stable than exact masks. For "blue eyes", late-layer amplitudes were nearly identical across seeds. For "add a rear spoiler", middle-layer amplitudes were consistent, but the gate sometimes shifted between trunk edge and rear window support. This indicates that layer selection is easier to learn than exact spatial placement for additive geometry. It may be useful in future work to combine layerwise residual gating with a user-provided point, box, or rough scribble.

The optional 20-step inference refinement improved directional CLIP by 2.4% and worsened outside-region LPIPS by 0.006. It helped difficult prompts such as "convert to a convertible" and "add wall art", but it slightly harmed clean part edits such as "blue eyes". The refinement stage adjusts residual amplitudes and gate temperatures while keeping gate-head weights fixed. Because it uses the test prompt and image, it behaves like per-image optimization rather than pure feedforward editing. The main paper does not include it in headline comparisons. Still, the result is useful: LSRG can be made more aggressive at inference time, but the cost is a predictable loss of preservation.

Failure analysis was performed on 300 manually inspected outputs. About 34% of failures were under-edits, where the prompt was only weakly expressed. About 27% were boundary errors, where the edit leaked into adjacent regions.

About 19% were generator-limit errors, where the requested object or attribute was poorly represented by the generator prior. About 11% were inversion errors amplified by the edit, and about 9% were semantic misunderstandings from the text encoder. The distribution differs by domain. Under-edits dominate faces because the method is conservative. Generator-limit errors dominate indoor scenes because the generator prior is less aligned with open-ended object additions.

A practical reliability score was computed by combining alignment and preservation thresholds. An edit was counted as acceptable if directional CLIP improved by at least 8%, outside-region LPIPS remained below 0.075, and identity similarity for faces remained above 0.84. Under this criterion, LSRG full accepted 78% of face edits, 69% of car edits, and 51% of indoor edits. The mapper baseline accepted 61%, 54%, and 43%, respectively. Instruction-tuned diffusion accepted 58%, 62%, and 57%. The thresholds are somewhat arbitrary, but they show that the preferred method depends on the domain and on whether preservation is mandatory. LSRG is strongest when the edit has a localized support and when preserving the original image is a first-order requirement.

VIII. Limitations and Ethical Considerations

The first limitation is generator dependence. LSRG cannot edit outside the representational capacity of the pretrained generator. If the generator has never learned a convincing version of a requested attribute, the residual controller has little material to work with. This is visible in indoor-scene object additions and in car edits requiring unusual geometry. Diffusion models have a broader open-world prior and can synthesize objects that a domain-specific GAN does not represent. The cost is that diffusion editors may resample or reinterpret larger parts of the image. LSRG occupies the opposite side of the trade-off: it preserves more reliably, but it cannot generate everything that language can describe.

The second limitation is inversion dependence. Real-image editing begins with an approximation $\hat{x}$ of the input image. If $\hat{x}$ loses jewelry, hair wisps, small logos, or unusual lighting, the editor may preserve the reconstruction rather than the original image. Feature blending reduces additional drift after inversion, but it does not recover details absent from the latent code. Encoder-only inversion makes the pipeline fast but less faithful. Optimization-refined inversion improves fidelity but adds substantial runtime. This bottleneck is not unique to LSRG, yet it is especially visible because the method is designed to preserve fine details.

The third limitation is prompt ambiguity. A text prompt rarely specifies all constraints needed for a unique local edit. “Add a lamp” does not say where the lamp should go. “Make the room warmer” can refer to color temperature, lighting, decor, or mood. “Make the car sporty” may imply paint, body kit, wheels, stance, or background. LSRG can only infer a plausible interpretation from training data and the generator prior. The model would be more predictable with user-provided spatial hints, but the present study intentionally evaluates text-only editing. A practical interface should probably combine text with optional points, masks, or sliders rather than relying on language alone.

The fourth limitation is metric dependence. Directional CLIP, LPIPS, identity cosine similarity, and Clean-FID capture different aspects of edit behavior, but none of them fully represents user judgment. CLIP can miss spatial mistakes and may reward stereotypical visual changes. LPIPS can penalize desired texture changes if the evaluation mask is imperfect. Identity embeddings can be sensitive to expression or occlusion. Clean-FID is distributional and not edit-specific. The paper therefore interprets metrics jointly and includes simulated preference judgments. Even then, a real deployment would require task-specific user studies, especially for creative editing where preferences differ.

The fifth limitation is social and ethical. Text-guided editing can be used for benign creative work, accessibility, design iteration, and image restoration, but it can also be used to create misleading imagery or to alter sensitive attributes without consent. Face editing deserves particular care because identity, age appearance, expression, and perceived demographic attributes can be changed with small prompts. The method’s conservative preservation behavior does not remove this risk. In some cases, better preservation may make manipulated images more convincing because the background and identity cues remain stable. Any deployment should include provenance signals, user education, abuse monitoring, and constraints around non-consensual manipulation.

Bias is also inherited from the pretrained generator and the image-text encoder. If the generator represents some identities, hairstyles, skin tones, or cultural attributes better than others, the editor will show uneven quality. If the text encoder associates prompts with biased visual patterns, residual directions may reproduce those associations. For example, prompts related to makeup, age, professionalism, or emotion can carry social stereotypes. The training protocol used here does not solve those problems. It only regularizes spatial and residual behavior. Bias evaluation should be conducted separately with balanced datasets and with prompts designed to reveal uneven edit quality across groups.

The method also raises an interpretability concern. Per-layer gates appear interpretable, but they are not semantic explanations in a strict sense. A gate shows where the model allowed feature changes, not why the text encoder considered the result aligned. A small gate can still lead

to downstream changes through generator dependencies. A residual amplitude can indicate layer participation, but it does not map directly to human concepts such as texture or shape. Layer traces should therefore be treated as diagnostic artifacts, not as complete explanations of model reasoning. Overstating their meaning would make the system seem more transparent than it is.

Finally, the fabricated experimental results in this paper are meant to define a coherent research study and plausible empirical behavior, not to replace a real benchmark. Exact values would depend on implementation details, datasets, baselines, hyperparameters, and human evaluation protocol. The methodological claims are narrower than the numbers: coupling residual strength with gate uncertainty is a plausible mechanism for improving locality; multi-layer blending is useful when prompts require both shape and texture changes; and preservation should be measured separately from prompt alignment. These claims are testable, and a real implementation should report confidence intervals, released code, and failure cases.

IX. Conclusion

This paper described Layerwise Semantic Residual Gating, a text-guided image editing framework that treats semantic editing as a coupled problem of residual selection, layer participation, and spatial feature preservation. The method uses a frozen generator and a compact prompt-conditioned controller to predict per-layer latent residuals and soft feature gates. Instead of relying only on a global latent displacement or a single manually chosen feature layer, it allows several generator resolutions to participate while making each residual accountable to a spatial support. The main technical mechanism is the residual-gate coupling loss, which penalizes large residual energy when gates remain uncertain. This encourages edits that are strong enough to satisfy the prompt but less likely to leak into unrelated regions.

The fabricated experiments suggest that this mechanism is most useful when the requested edit is localized and when preserving unmentioned content is central to the task. On face and car prompts, LSRG improved directional text alignment relative to latent baselines while reducing outside-region perceptual drift. Against diffusion editors, it generally preserved identity and background more strongly, although diffusion methods often produced more forceful semantic changes. This trade-off should not be collapsed into a single score. The better method depends on whether the user wants a restrained edit to an existing image or a more generative reinterpretation of the scene.

The ablations indicate that locality does not come only from predicting a mask. When residual norm and mask area were regularized independently, edits still leaked through uncertain gates and downstream generator dependencies. Entropy annealing also mattered because early hard masks caused incomplete edits. The compact segment-selection variant showed that fixed semantic regions can work well when the prompt vocabulary matches the segmentation vocabulary, but learned gates were more flexible for soft boundaries, textures, and prompts spanning several visual components. These results point toward a broader principle: editing systems should regulate the strength, place, and resolution of changes jointly rather than treating them as separate implementation details.

Several limitations remain. The method depends on generator editability, inversion quality, and the semantic reliability of the image-text encoder. It struggles with prompts requiring occluded content, exact object placement, or broad illumination changes. It also inherits biases and misuse risks from the generative and contrastive models on which it depends. The gate maps offer useful diagnostics but should not be interpreted as complete semantic explanations. A practical system would benefit from optional user spatial guidance, stronger grounding models, and explicit provenance mechanisms.

Future work should test residual-gate coupling in latent diffusion models, where attention maps and denoising features may provide a different layer hierarchy. Another direction is interactive editing, where the user can accept, reject, or adjust proposed gates before the residual is applied. A third direction is compositional prompt parsing, allowing separate residual-gate fields for separate clauses such as color, object addition, and expression. The present study keeps those problems outside its scope. Its narrower conclusion is that localized text-guided editing becomes more reliable when the editor is trained to justify latent movement through spatially and hierarchically constrained feature changes.

REFERENCES

[1] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in Neural Information Processing Systems*, vol. 27, pp. 2672–2680, 2014.

[2] T. Karras, S. Laine, and T. 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.

[3] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. 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.

[4] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark,

G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proceedings of the 38th International Conference on Machine Learning*, vol. 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763, PMLR, 2021.

[5] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022.

[6] R. Abdal, Y. Qin, and P. Wonka, "Image2stylegan: How to embed images into the stylegan latent space?," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4432–4441, 2019.

[7] R. Abdal, Y. Qin, and P. Wonka, "Image2stylegan++: How to edit the embedded images?," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8296–8305, 2020.

[8] Y. Shen, C. Yang, X. Tang, and B. Zhou, "Interpreting the latent space of gans for semantic face editing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9243–9252, 2020.

[9] E. Härkönen, A. Hertzmann, J. Lehtinen, and S. Paris, "Ganspace: Discovering interpretable gan controls," in *Advances in Neural Information Processing Systems*, vol. 33, pp. 9841–9850, 2020.

[10] Z. Wu, D. Lischinski, and E. Shechtman, "Stylespace analysis: Disentangled controls for stylegan image generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12863–12873, 2021.

[11] A. Revanur, D. Basu, S. Agrawal, D. Agarwal, and D. Pai, "Coralstyleclip: Co-optimized region and layer selection for image editing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12695–12704, 2023.

[12] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Advances in Neural Information Processing Systems*, vol. 33, pp. 6840–6851, 2020.

[13] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," in *International Conference on Learning Representations*, 2021.

[14] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," in *Advances in Neural Information Processing Systems*, vol. 34, pp. 8780–8794, 2021.

[15] A. Nichol, P. Dhariwal, A. Ramesh, P. Shyam, P. Mishkin, B. McGrew, I. Sutskever, and M. Chen, "Glide: Towards photorealistic image generation and editing with text-guided diffusion models," 2021.

[16] T. Brooks, A. Holynski, and A. 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.

[17] T. Karras, M. Aittala, S. Laine, E. Härkönen, J. Hellsten, J. Lehtinen, and T. Aila, "Alias-free generative adversarial networks," in *Advances in Neural Information Processing Systems*, vol. 34, pp. 852–863, 2021.

[18] E. Richardson, Y. Alaluf, O. Patashnik, Y. Nitzan, Y. Azar, S. Shapiro, and D. Cohen-Or, "Encoding in style: A stylegan encoder for image-to-image translation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2287–2296, 2021.

[19] O. Tov, Y. Alaluf, Y. Nitzan, O. Patashnik, and D. Cohen-Or, "Designing an encoder for stylegan image manipulation," *ACM Transactions on Graphics*, vol. 40, no. 4, pp. 1–14, 2021.

[20] E. Collins, R. Bala, B. Price, and S. Süsstrunk, "Editing in style: Uncovering the local semantics of gans," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5771–5780, 2020.

[21] O. Patashnik, Z. Wu, E. Shechtman, D. Cohen-Or, and D. Lischinski, "Styleclip: Text-driven manipulation of stylegan imagery," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2085–2094, 2021.

[22] U. Kocasari, A. Dirik, M. Tiftikci, and P. Yanardag, "Stylemc: Multi-channel based fast text-guided image generation and manipulation," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3441–3450, 2022.

[23] R. Gal, O. Patashnik, H. Maron, G. Chechik, and D. Cohen-Or, "Stylegan-nada: Clip-guided domain adaptation of image generators," *ACM Transactions on Graphics*, vol. 41, no. 4, pp. 1–13, 2022.

[24] A. Revanur, D. D. Basu, S. Agrawal, D. Agarwal, and D. Pai, "Mask conditioned image transformation based on a text prompt," Mar. 5 2026. US Patent App. 19/377,653.

[25] O. Avrahami, D. Lischinski, and O. Fried, "Blended diffusion for text-driven editing of natural images," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18208–18218, 2022.

[26] A. Hertz, R. Mokady, J. Tenenbaum, K. Aberman, Y. Pritch, and D. Cohen-Or, "Prompt-to-prompt image editing with cross attention control," 2022.

[27] R. Mokady, A. Hertz, K. Aberman, Y. Pritch, and D. 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.

[28] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick, "Segment anything," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4015–4026, 2023.

[29] Y. Li, H. Liu, Q. Wu, F. Mu, J. Yang, J. Gao, C. Li, and Y. J. Lee, "Gligen: Open-set grounded text-to-image generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22511–22521, 2023.

[30] L. Zhang, A. Rao, and M. Agrawala, "Adding conditional control to text-to-image diffusion models," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3813–3824, 2023.

[31] N. Ruiz, Y. Li, V. Jampani, Y. Pritch, M. Rubinstein, and K. Aberman, "Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22500–22510, 2023.