This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

FDS: Frequency-Aware Denoising Score for Text-Guided Latent Diffusion Image Editing

Yufan Ren[1] Zicong Jiang[1,2,3] Tong Zhang[1†] Søren Forchhammer[2] Sabine Süsstrunk[1]
[1]School of Computer and Communication Sciences, EPFL
[2]Department of Electrical and Photonics Engineering, DTU
[3] Department of Electrical Engineering, Chalmers University of Technology

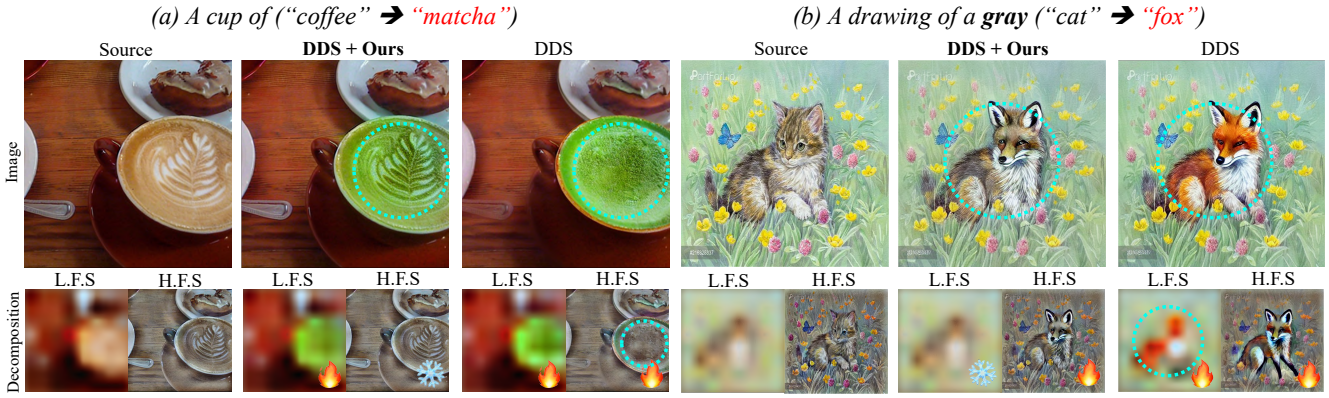


Figure 1. Text-guided image editing using Text-to-Image (T2I) models, such as DDS [17], often fails to produce satisfactory results due to **indiscriminate optimization across all frequency subbands**. For example, in the top row, DDS removes the detailed pattern of latte art (a) and drastically alters the cat's color despite "gray" being specified in the prompt (b). These issues become more apparent through frequency decomposition during optimization (L.F.S and H.F.S refer to low frequency subband and high frequency subband, respectively) in the second row, where unnecessary modifications occur. Our method selectively optimizes frequency bands, preserving high-frequency details in the latte art (a) and maintaining color consistency in the gray cat (b). The "freeze" symbol indicates frozen frequency components, while the "flame" means optimized. Best viewed on a screen when zoomed in.

Abstract

Text-guided image editing using Text-to-Image (T2I) models often fails to yield satisfactory results, frequently introducing unintended modifications, such as the loss of local detail and color changes. In this paper, we analyze these failure cases and attribute them to the indiscriminate optimization across all frequency bands, even though only specific frequencies may require adjustment. To address this, we introduce a simple yet effective approach that enables the selective optimization of specific frequency bands within localized spatial regions for precise edits. Our method leverages wavelets to decompose images into different spatial resolutions across multiple frequency bands, enabling precise modifications at various levels of detail. To extend the applicability of our approach, we provide a comparative analysis of different frequency-domain techniques. Additionally, we extend our method to 3D texture editing by performing frequency decomposition on the triplane representation, enabling frequency-aware adjustments for 3D textures. Quantitative evaluations and user studies demonstrate the effectiveness of our method in producing high-quality and precise edits. Further details are available on our project website: https://ivrl.github.io/fds-webpage/

1. Introduction

Recent advances in Text-to-Image (T2I) models, such as DALL-E 2 [39] and Latent Diffusion Models (LDMs) [40], have significantly advanced the field of text-based im-

†: Corresponding author.

age generation. Despite these advances, achieving precise text-guided image editing with T2I models remains challenging. Existing methods, including DreamFusion [37] with its Score Distillation Sampling (SDS) technique and subsequent enhancements such as Delta Denoising Score (DDS) [17] and Contrastive Distillation Score (CDS) [34], have made notable progress. However, they still struggle with fine-grained control over edits. In particular, they often fail to preserve high-frequency details or alter the color, leading to unsatisfactory results in tasks that require precise modifications. For example, as shown in Fig. 1, when DDS is used to transform "a cup of **coffee**" into "a cup of **matcha**", intricate latte art details are often unintentionally removed; converting a drawing of "a gray cat" into "a gray fox" changes the cat's color to reddish-brown, likely due to data bias, despite specifying the color "**gray**" in the prompt.

We identify that these limitations arise from two primary challenges. First, language-vision models struggle to disentangle complex attributes [1, 29, 38], which hinders the faithful preservation of source image attributes that should remain unchanged (e.g., the fox's color in Fig. 1). However, fundamentally improving this aspect is resource-intensive and may require retraining a more robust text-image model with better annotated paired data. Second, and more directly, T2I editing models are hindered by indiscriminate optimization across all frequency subbands during editing, often causing unintended alterations (both cases in Fig. 1). Therefore, providing additional levels of controllability in the frequency domain for text-guided editing is needed.

In this work, we propose a simple yet effective approach that enhances text-guided image editing by revisiting classical frequency domain techniques. We introduce a method that decomposes images into frequency subbands [11], enabling selective optimization of each. We showcase the benefit of frequency-awareness for editing in two cases: local detail preservation and color fidelity preservation. This selective optimization provides more precise control over the editing process and leads to more satisfying results. Additionally, we extend our method to 3D texture editing and propose a frequency-aware texture editing method that uses a frequency-decomposed triplane representation [5]. This extension enhances frequency awareness, improving both color fidelity and the preservation of high-frequency details in textures. Our contributions are as follows:

- We identify issues with local detail preservation and color fidelity in score distillation-based image editing methods, and attribute these issues to indiscriminate optimization across all frequency subbands.
- We propose a novel text-guided image editing method that leverages frequency domain techniques to provide precise control within the frequency domain.
- We demonstrate that selectively optimizing specific frequency subbands leads to better preservation of image details and color fidelity compared to baseline methods.
- We introduce a frequency-decomposed triplane representation tailored for 3D texture editing.
- We provide an analysis of different frequency domain techniques and offer a guideline on their application to the text-guided image editing task.

2. Related Work

Text-to-Image Models. Text-to-Image (T2I) models [20], such as DALL-E 2 [39], Stable Diffusion [40], and Imagen [41], have seen remarkable success in generating high-quality images from text prompts. Due to their powerful diffusion-based generative priors, T2I models have been widely applied in various domains including depth estimation [22], image-to-image translation [10, 30], image super-resolution [33, 44], and image inpainting [8, 35, 46].

Image Editing with Text-to-Image Models. Recent research has explored various applications of T2I models in image editing, including interacting with the reverse process of an inverted latent [4, 7, 16, 19, 21, 23, 25, 31, 32] or fine-tuning the diffusion model to learn the reverse process to restore the edited image, such as Instruct-Pix2Pix [2]. While many of these methods exploit the reverse process of diffusion models, a different method known as Score Distillation Sampling (SDS) [26, 37], originally proposed for 3D object generation, has shown promising performance. However, despite SDS being effective in generating 3D assets, its gradient updates can introduce noise and result in imprecise edits, particularly when applied to fine image editing tasks. Delta Denoising Score (DDS) [17] is a method that improves upon SDS by canceling out noisy gradients and focusing on the desired modifications in a more localized manner, leading to better preservation of image details during editing. CDS [34] leverages a constrastive loss term to enforce that the semantics at each spatial position remain the same during optimization.

Our approach is compatible with score distillation-based methods, integrating wavelet decomposition as an image representation to enable selective modification of certain frequency components during editing.

Frequency Domain Techniques in Image Editing. Despite frequency domain techniques having been deployed in various ways of image manipulation [3, 12, 13, 27, 47–49], their application to text-guided image editing is still limited. FlexiEdit [25] adds randomness to high frequency components during diffusion inversion allowing for non-rigid editing. FreeDiff [45] emphasizes high frequency component during the diffusion sampling process. [14] maps frequency information to images using a ControlNet [50] fashion.

Differently, our method leverages Wavelet Transforms [11] in fine-grained image editing with score distillation sampling, enabling the preservation of detail and color fidelity. Besides, similar to other score distillation methods,

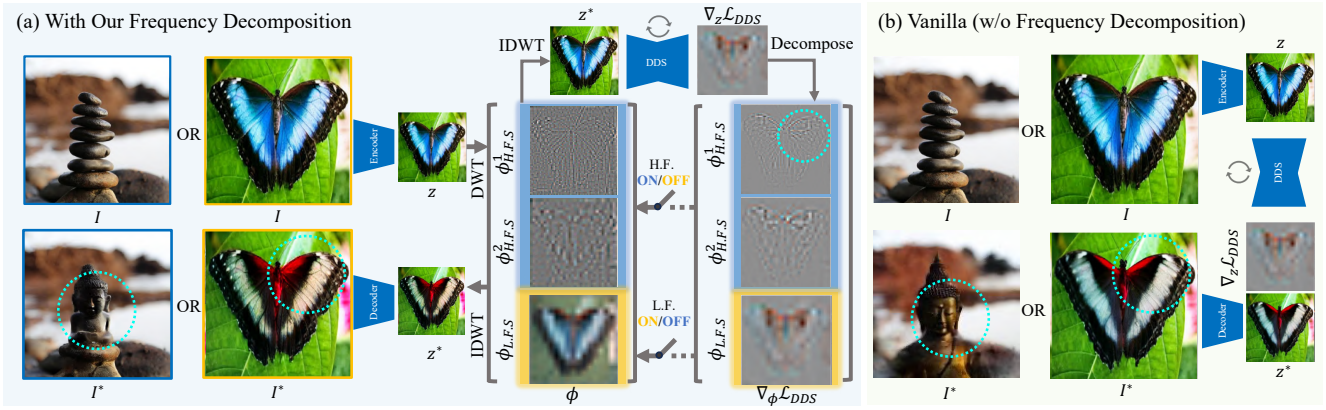


Figure 2. **Method overview.** Ours (a) differs from vanilla score distillation editing (b), which backpropagates gradients to the latent space (z)to perform editing. Our method leverages wavelet frequency decomposition to decompose latent z into wavelet subbands ϕ including high frequency ($\phi_{\text{H.F.S}} = \{\phi^1_{\text{H.F.S}}, \phi^2_{\text{H.F.S}}, \cdots, \phi^J_{\text{H.F.S}}\}$) and low frequency ($\phi_{\text{L.F.S}}$). We process the reconstructed latent z^* with the diffusion model to obtain a gradient for optimization, which is applied to either high-frequency components or low frequency components selectively depending on application. Consequently, our method produces edits that benefit from detail preservation (butterfly case, **yellow** for text and image borders) and color fidelity (stone, **blue** for text and image borders). Best viewed on a screen when zoomed in.

our method does not require training or finetuning diffusion models.

3. Frequency-Aware Score Distillation

Our work enhances SDS [37] and its improved versions, e.g., DDS [17] and CDS [34], by incorporating Wavelet Transform [11] to enable selective frequency optimization. Unlike the purely spatial approach used in standard representations, our method employs a spatial-frequency representation and seamlessly integrates within the score distillation frameworks. Without loss of generality, we demonstrate our approach in image editing based on DDS and texture editing based on SDS as they achieve the best results.

3.1. Frequency-Decomposed Image Editing

3.1.1. Initial Setup and DDS

Given a source image I of shape $3 \times 512 \times 512$, we encode it via a Variational Autoencoder (VAE) [24] to obtain its latent representation z with dimensions $4 \times 64 \times 64$. In the original DDS approach, optimization is performed directly on z by calculating the loss gradient $\nabla_z \mathcal{L}_{\text{DDS}}$ (conditioned on a source prompt and a target prompt) and iteratively updating z over N steps before decoding it at the end.

3.1.2. Wavelet Decomposition and Subband Extraction

Unlike the vanilla DDS, we enable selective optimization by decomposing the latent z into multiple frequency subbands ϕ through DWT [11]. We set the decomposition level J and select an appropriate wavelet type, such as Daubechies wavelets [11], which will be analyzed in Sec. 5.

For latent visualization, we use a linear transform from four-channel latent to three-channel RGB.

The decomposition yields subbands ϕ, consisting of a low-frequency component $\phi_{\text{L.F.S}}$ and multiple high-frequency components $\phi_{\text{H.F.S}} = \{\phi^1_{\text{H.F.S}}, \phi^2_{\text{H.F.S}}, \ldots, \phi^J_{\text{H.F.S}}\}$, where each level of high-frequency subbands contains three directional details, such as $\phi^1_{\text{H.F.S}_{HL}}$, $\phi^1_{\text{H.F.S}_{LH}}$, and $\phi^1_{\text{H.F.S}_{HH}}$. The low-frequency subband $\phi_{\text{L.F.S}}$ has dimensions $4 \times \frac{64}{2^J} \times \frac{64}{2^J}$.

3.1.3. Selective Optimization

Using the Inverse Discrete Wavelet Transform (IDWT), we reconstruct $\hat{z}$ from the subbands ϕ which is then input into the diffusion model to compute the gradient $\nabla_\phi \mathcal{L}_{\text{DDS}}$, which aims to modify the target attributes.

The gradient $\nabla_\phi \mathcal{L}_{\text{DDS}}$ propagates back to each subband, including both $\phi_{\text{L.F.S}}$ and $\phi_{\text{H.F.S}}$. As these subbands are categorized into low-frequency components $\phi_{\text{L.F.S}}$ and high-frequency components $\phi_{\text{H.F.S}}$, we use a stop-gradient operation during optimization (denoted as StopGrad$(\cdot)$) for selective gradient control. If the objective is to preserve low-frequency content (e.g., color), we apply the StopGrad to $\phi_{\text{L.F.S}}$ to prevent updating to it and focusing optimization on $\phi_{\text{H.F.S}}$. Conversely, if preserving high-frequency details is desired, we apply StopGrad to $\phi_{\text{H.F.S}}$ to restrict updates to the low-frequency components only. After N optimization iterations, we decode the optimized latent $\hat{z}$ to produce the final output image. This output image preserves the original attributes intended for retention, with targeted adjustments as specified by the prompts.

3.2. Frequency-Aware 3D Texture Editing

We extend our method to 3D texture editing on meshes, aiming to modify textures while preserving detail and color fidelity (Fig. 3).

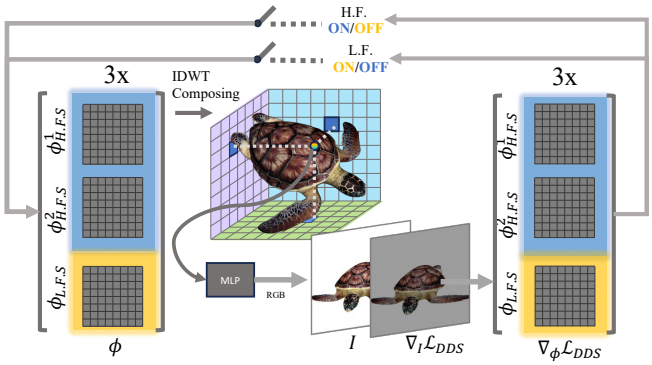


Figure 3. **3D texture editing pipeline**. We represent a 3D texture field as a frequency-decomposed triplane ϕ, i.e., three sets of wavelet subbands representing $XY - YZ - XZ$ in three directions. To render an image at camera view p, we construct a triplane from ϕ, which is queried for colors. The rendered image is processed by the latent diffusion model to produce a gradient, which is backpropagated to update selected frequency components.

3.2.1. Frequency-Aware Triplane Representation

Given a mesh $\mathbf{M}$ with triangular faces, we denote its original texture before editing as $\mathbf{T}$. To represent the texture field to be optimized, we use a triplane [6] with three planes P_{xy}, P_{xz}, and P_{zy}, each sized $C \times H \times W$. Rendering an image from the mesh $\mathbf{M}$ requires retrieving the color of each pixel. To do so, for each pixel, we cast a ray from the camera and calculate the ray-mesh intersection $\mathbf{x} = [x, y, z]$ (where $x, y, z \in [-1, +1]$ in a normalized space). To retrieve the color at this point $\mathbf{x}$, we project it onto these planes using bilinear interpolation, which yields three C-channel vectors. These three vectors are combined to form a $3 \times C$ feature vector, which is then decoded into RGB values using an MLP.

To incorporate frequency awareness, similar to 2D image editing, we represent each plane in the triplane using a set of wavelet subbands, Fig. 3. Before rendering, we apply the IDWT to reconstruct the triplane from subbands, i.e., $\hat{P} = f_{\text{IDWT}}(\phi_{\text{H.F.S}}, \phi_{\text{L.F.S}})$ for each dimension and use the reconstructed triplane $\hat{P}$ in the rendering process described earlier. Before editing, we initialize the triplane and MLP by fitting the rendered image to the original texture map $\mathbf{T}$ using the L1 loss $\mathcal{L}_1$.

3.2.2. Selective Optimization

After rendering the image I from frequency-aware triplane, we use a VAE to encode it into latent z. We then apply a diffusion model to compute the score distillation loss $\mathcal{L}_{\text{SDS}}$ and we obtain the gradient $\nabla_\phi \mathcal{L}_{\text{SDS}}$ by backpropagation. Similar to 2D image editing, we apply StopGrad$(\cdot)$ operator to either the low- or high-frequency components.

3.3. Implementation Details

Our implementation leverages PyTorch [36], with wavelet transformations handled by the differentiable wavelet library `wavelet_pytorch` [9]. Similar to [17], we use Stable Diffusion [40] as the image diffusion model. Moreover, we conduct experiments on an NVIDIA RTX 3090 GPU. We set the number of iterations to $N = 500$ for score distillation sampling.

4. Experiments

In this section, we evaluate the effectiveness of our method in both 2D image editing and 3D texture editing. We begin by introducing our baseline methods and experimental protocols. Next, we report qualitative and quantitative comparisons. Finally, we present the results from a user study.

4.1. Experimental Settings

4.1.1. 2D Image Editing

Baseline Methods. To verify the effectiveness of our method, we conducted comparative experiments in 2D. Since our method is optimization-based, we primarily compare it with state-of-the-art score distillation methods, including DDS [17] and CDS [34], as these are the most competitive baselines. Moreover, we report results comparing various diffusion sampling methods, including Plug-and-Play (PnP) [43], Prompt-to-Prompt (P2P) [16], DiffuseIT [28], FlexiEdit [25], and FreeDiff [45]. We also report on DreamSampler [23], which unifies diffusion sampling and score distillation.

Selecting Frequency Components. As mentioned previously, we showcase our method's capability in two primary editing scenarios: color editing and detail editing, where we optimize the low-frequency and high-frequency components. Since selecting specific frequency components for optimization is a unique advantage of our method—one not supported by baseline methods, we compare with the baseline methods using their closest settings, such as the style modification in FreeDiff [45], when applicable.

Evaluation Protocol. For quantitative results, we sample 200 cat images from the LAION-COCO dataset [42] and use the editing methods to transform them into dogs, pigs and cows. We use the Structural Similarity Index Measure (SSIM) to assess fidelity and LPIPS [51] to quantify the preservation of high-frequency details. We also measure the CLIP score [18] to evaluate alignment between edits and target text descriptions.

4.1.2. 3D Texture Editing

We extend our method directly from 2D to 3D texture editing to demonstrate the benefits of frequency disentangle-

We provide additional experiment results beyond these two primary settings, such as combing ours with FlexiEdit [25] to offer non-rigid editing capability, in supplementary material.



Figure 4. **Qualitative results.** We conducted a qualitative comparison with the most competitive baselines. For low-frequency editing, our method follows instructions closely while preserving high-frequency details better. In the first row, stone lion, our method preserves details of the lion's eyes and mouth (**A**). On the contrary, CDS, DDS and other methods lose these structures, introducing significant changes. For the second row, the chicken, we preserve the beak and eye areas (**B**). In contrast, other methods distort the structure noticeably or fail to generate meaningful images (e.g., DiffuseIT) and follow the target description (DreamSampler, FlexiEdit). CDS, the best among baselines, alters the beak. For high-frequency editing, our method maintains better color fidelity than the baselines. In the first row, our approach preserves color consistency in the transformation from cat to fox. In the stone-to-Buddha case, our method preserves both the background and statue colors (**C**) better than CDS and similar methods. In the third row, our method preserves image color information better, especially the pupil and face skin color, while still modifying details (**D**). Other methods introduce structure distortion, which can be attributed to the lack of global information guidance. Best viewed on a screen when zoomed in.

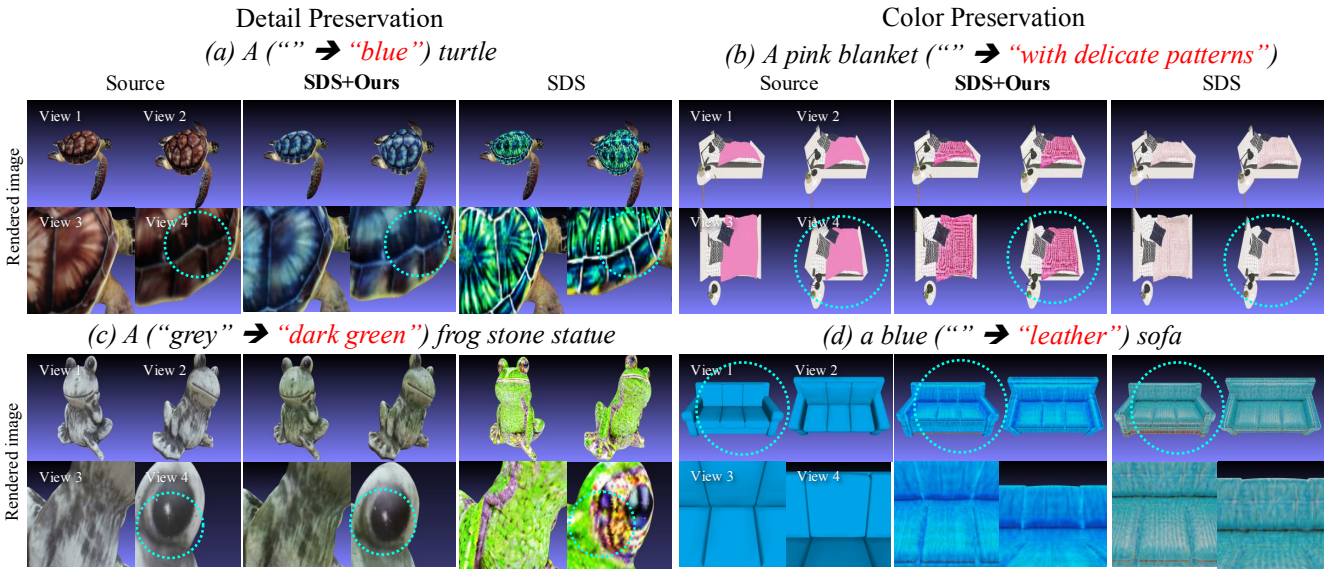


Figure 5. **Qualitative results.** Our frequency-aware denoising score method on 3D texture. We compare our results with the pure triplane representation. In the detail preservation cases (a) and (c), our method faithfully preserves the original texture from the source texture map. In contrast, SDS distorts the original texture and does not follow the text "blue" in the turtle case. In the color fidelity cases (b) and (d), our method retain the pink color of the quilt and the blue color of the sofa, while adding intricate patterns and texture details guided by the text. However, SDS creates patterns on the quilt but changes the main color to a much lighter pink. In the sofa case, SDS also adds texture details but shifts the main color to a greenish-blue hue whitish-red hue. Best viewed on a screen when zoomed in.

ment offered by our approach. Firstly, since diffusion sampling methods [16] cannot be applied to 3D texture editing at inference time, we experiment with score distillation methods only. Secondly, we observe that some score distillation methods such as DDS and CDS do not perform well in 3D texture editing, we limit our comparison to SDS, to test the effect of our frequency-aware denoising score. Similar to the 2D tasks, we examine performance differences in both low-frequency color editing and high-frequency detail editing tasks.

4.2. 2D Image Editing Results

We present qualitative results in Fig. 4 with additional results provided in the supplementary material. In the detail preservation case (upper part), none of the baseline methods are able to faithfully preserve the geometry of the stone lion's face and mouth or the chicken's eye and beak. Among score distillation methods, CDS removes the lion's teeth and changes the size of the chicken's eye. DDS, on which our method is based, also removes the lion's teeth and alters the size of its eye. Diffusion sampling methods often significantly change the structure and may fail to generate meaningful results, e.g., DreamSampler [23] and DiffuseIT [28].

In the color fidelity case (lower part), DDS and CDS often disrupt color consistency by adding white to the gray cat or turning the stone yellowish. Diffusion sampling methods frequently fail to preserve color, as seen in the cat-to-fox and shark cases with PnP. In contrast, since our method explicitly disentangles high-frequency and low-frequency components and optimizes the selected frequency band, our method achieves better color consistency.

For the quantitative evaluation, we report the results against the most competitive baselines, CDS and DDS, in Tab. 1. Our method achieves the same CLIP score as CDS, indicating that our method performs as well as CDS in matching the text guidance. Additionally, our method achieves slightly better scores on LPIPS and SSIM, indicating improved structural preservation and consistency.

Method		CLIP↑	LPIPS ↓	SSIM ↑
Diffusion Sampling	NMG [7]	**23.68**	0.182	0.738
	Direct Inversion [19]	23.61	0.284	0.676
Score Distillation	DDS [17]	23.64	0.297	0.736
	CDS [34]	23.65	0.136	0.818
	Ours	23.65	**0.129**	**0.819**

Table 1. **Quantitative results on 2D image editing**. Our method achieves comparable text alignment (CLIP score) with CDS while outperforming DDS and CDS in both SSIM and LPIPS.

In Tab. 2, we present the user study results. We surveyed 24 users using six source images, each accompanied by two edited versions. Half of these images focused on detail preservation, and the other half on color fidelity. Each user was shown a set of three images: the source image, an edit labeled as option one, and another edit labeled as option two. We asked each user to select the preferred option between the two edits that better "reflects the specified

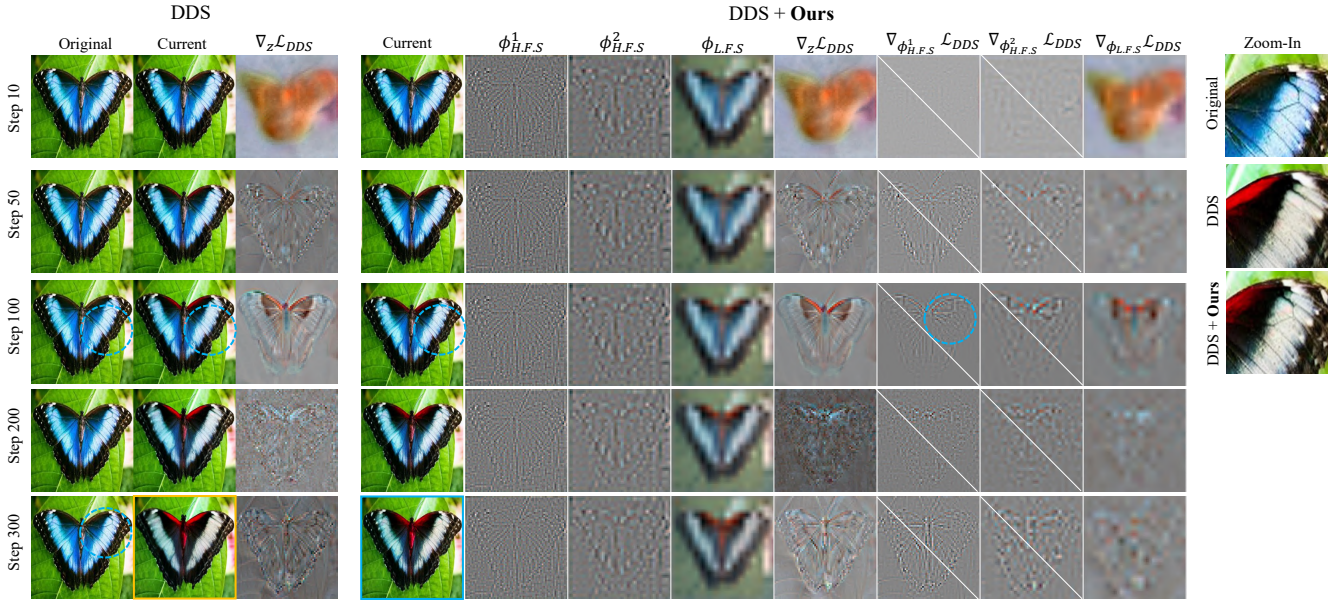


Figure 6. **Gradient visualization during optimization.** Edit with "A photo of a (blue → red) butterfly". We visualize the frequency-decomposed representation ϕ (with three subbands, $\phi^1_{\text{H.F.S.}}$, $\phi^2_{\text{H.F.S.}}$ and $\phi_{\text{L.F.S.}}$), the gradients at the latent level $\nabla_z \mathcal{L}_{\text{DDS}}$, as well as gradients at subband level $\nabla_\phi \mathcal{L}_{\text{DDS}}$ in multiple wavelet subbands. As shown, although the optimization points towards the target, the gradients at each step exhibit randomness ($\nabla_z \mathcal{L}_{\text{DDS}}$ in step 50 and step 100). This is due to the noise added during the sampling procedure and tends to smooth out or distort detail patterns. With our method, we apply a stop-gradient operator to preserve these details (right). Note that we add a diagonal line to indicate the subband gradients where the stop-gradient operator is applied. Best viewed on a screen when zoomed in.

text instructions, while ensuring that local detail patterns or color remain unchanged". According to the results, on average, more than 80% of users prefer the edits made with our method over both CDS and DDS.

	Color pres.↑	Detail pres. ↑	Average
Prefer ours over DDS	73.8%	90.5%	82.1%
Prefer ours over CDS	92.9%	79.8%	86.3%

Table 2. **Percentage of users preferring ours over baselines** in two editing scenarios: color preservation and detail preservation. More users preferred our method in both cases. More than 90% users prefer ours in detail preservation than DDS, and color preservation than CDS. On average, over 80% of users favored the edits generated by our method over CDS and DDS.

4.3. Texture Editing Results

Similar to the 2D case, our method enables the preservation of texture details and color fidelity, Fig. 5. In detail preservation, our method can change the low-frequency components of the turtle and frog, while preserving the detailed textures. In contrast, the baseline SDS changes the shell texture and shifts the frog's eyes. In the color preservation setting, ours adds high-frequency texture to the quilt and adds leather texture to the sofa, while maintaining color consistency, whereas the baseline SDS alters the color.

5. Discussions

5.1. Gradient Visualization in Score Distillation

In Fig. 6, we visualize the gradient $\nabla_z \mathcal{L}_{\text{DDS}}$ in DDS. It is noteworthy that although the gradient generally aligns with the text guidance (e.g., turning the blue butterfly to red), the inherent randomness of score distillation causes inconsistencies in $\nabla_z \mathcal{L}_{\text{DDS}}$ across multiple steps. Consequently, this noise alters the detailed structure on the wing. By applying our method with a stop-gradient operation to specific frequency bands, we achieve the desired editing while preserving local details (right). We provide gradient visualization for high frequency editing in supplementary material.

5.2. Choosing Among the Wavelet Family

Different wavelets have distinct filters, which enable them to extract different types of features. In this experiment, we analyze the two attributes that most significantly impact the result: smoothness and decomposition level in the Daubechies wavelet family [11], Fig. 7. We use the instruction to turn "a coffee latte art" into "a matcha latte art" while editing the low-frequency component. Due to the halving nature of wavelets, a higher J value results in a lower resolution for the low-frequency subband. We observe reduced localization ability at high J levels (e.g., J=5), causing color to spill over into adjacent regions. Regarding smoothness, when DB=1, Daubechies wavelet reduces to

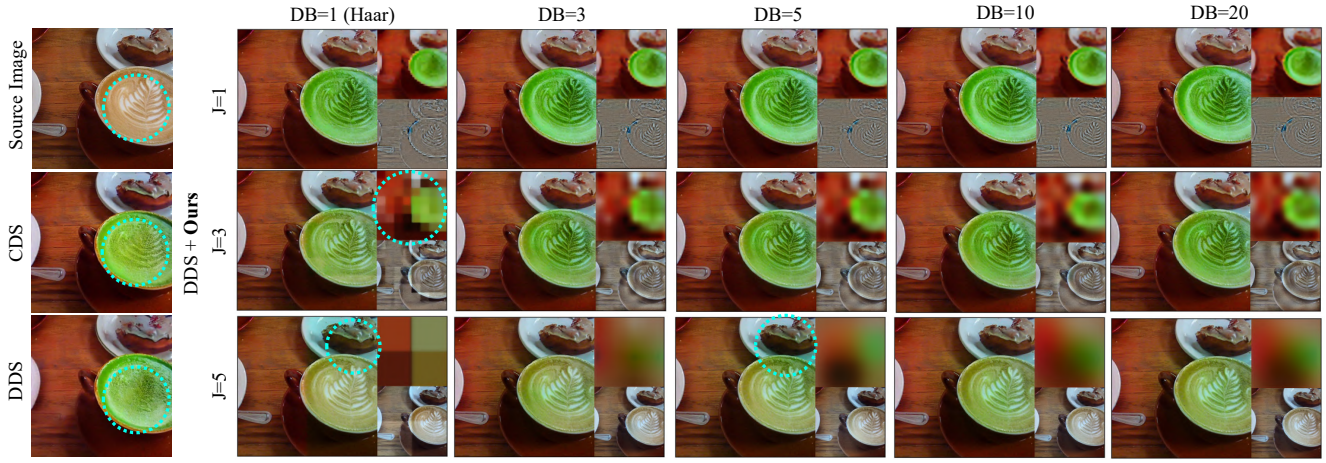


Figure 7. **Comparing different wavelets parameters**. Edit with "A cup of (coffee → matcha)". Rows: Daubechies wavelets with an increasing number of vanishing moments. Columns: levels in multi-level decomposition. Larger Daubechies wavelet indices result in smoother transformations, in contrast to the mosaic-like pattern created by the Haar Wavelet. Larger J values lead to lower-resolution low frequency components, making it difficult to model the round shape of the matcha cup at J=5. Best viewed on a screen when zoomed in.

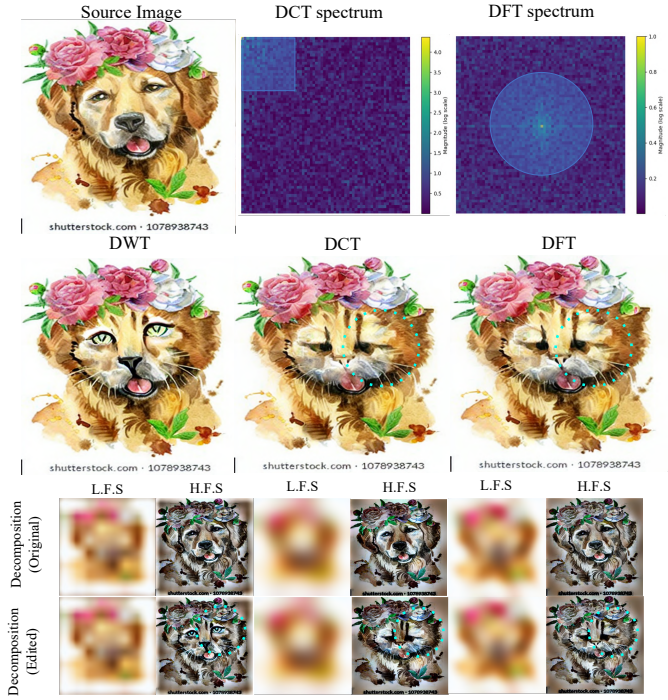


Figure 8. **Comparison of different frequency decomposition methods**. Edit with "A drawing of a (dog → cat). Both DCT and DFT lack spatial dimension in decomposition, causing artifacts like the cat's eyes. In the spectrum, the blue areas represent the mask schematic used in this experiment. For both DFT and DCT, we applied mask to the low-frequency regions to limit parameter updates. Best viewed on a screen when zoomed in.

the Haar wavelet [15], featuring sharp edges, while a larger index produces a smoother result. Generally, a medium setting (e.g., J=3 and DB=3) yields decent outputs.

5.3. Comparing Other Frequency Methods

Wavelets provide distinct advantages over other frequency decomposition methods, such as superior spatial localization and multiresolution decomposition capabilities. We compare wavelets with DFT and DCT in Fig. 8. Regarding the DFT and DCT parameters, we select a frequency threshold that achieves high-low frequency separation similar to ours (original row). Although all three decomposition methods enable frequency separation, already improving upon DDS, we find that the superior localization capability of wavelets makes them a better choice in certain cases.

5.4. Limitations

While our selective frequency optimization improves color fidelity and detail preservation for precise editing tasks, it inherits the limitations of diffusion models and score distillation, such as challenges in accurately interpreting prompts and requiring a lengthy sampling process

6. Conclusion

In this paper, we identify the issue of indiscriminate optimization across all frequency subbands in score distillation text-guided image editing. To address this, we propose a frequency-aware denoising score that utilizes the discrete wavelet transform to selectively optimize specific frequency subbands. This approach achieves improved preservation of high-frequency details and enhances color consistency. We hope our proposed method will pave the way for fine-grained, controllable image and texture editing.

7. Acknowledgment

Zicong Jiang is currently affiliated with Chalmers University of Technology. He made significant contributions to the code implementations and experimental validations during his master's thesis work at IVRL. The authors extend their gratitude to Quentin Bammey, Raphaël Achddou, Dongqing Wang, and Martin Everaert for their thorough proofreading and valuable feedback, as well as to the reviewers for their constructive suggestions.

References

[1] Yevgeni Berzak, Andrei Barbu, Daniel Harari, Boris Katz, and Shimon Ullman. Do you see what i mean? visual resolution of linguistic ambiguities. *arXiv preprint arXiv:1603.08079*, 2016. 2

[2] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *CVPR*, pages 18392–18402, 2023. 2

[3] Mu Cai, Hong Zhang, Huijuan Huang, Qichuan Geng, Yixuan Li, and Gao Huang. Frequency domain image translation: More photo-realistic, better identity-preserving. In *CVPR*, pages 13930–13940, 2021. 2

[4] Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In *ICCV*, pages 22560–22570, 2023. 2

[5] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *CVPR*, pages 16123–16133, 2022. 2

[6] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *CVPR*, pages 16123–16133, 2022. 4

[7] Hansam Cho, Jonghyun Lee, Seoung Bum Kim, Tae-Hyun Oh, and Yonghyun Jeong. Noise map guidance: Inversion with spatial context for real image editing. *arXiv preprint arXiv:2402.04625*, 2024. 2, 6, 1

[8] Ciprian Corneanu, Raghudeep Gadde, and Aleix M Martinez. Latentpaint: Image inpainting in latent space with diffusion models. In *WACV*, pages 4334–4343, 2024. 2

[9] Fergal Cotter. *Uses of complex wavelets in deep convolutional neural networks*. PhD thesis, 2020. 4

[10] Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance. *arXiv preprint arXiv:2210.11427*, 2022. 2, 1

[11] Ingrid Daubechies. *Ten lectures on wavelets*. SIAM, 1992. 2, 3, 7

[12] Iddo Drori, Daniel Cohen-Or, and Hezy Yeshurun. Fragment-based image completion. *ACM Trans. Graph.*, 22 (3):303–312, 2003. 2

[13] Martin Nicolas Everaert, Athanasios Fitsios, Marco Bocchio, Sami Arpa, Sabine Süsstrunk, and Radhakrishna Achanta. Exploiting the signal-leak bias in diffusion models. In *WACV*, pages 4025–4034, 2024. 2

[14] Xiang Gao, Zhengbo Xu, Junhan Zhao, and Jiaying Liu. Frequency-controlled diffusion model for versatile text-guided image-to-image translation. In *AAAI*, pages 1824–1832, 2024. 2

[15] Alfred Haar. Zur theorie der orthogonalen funktionensysteme. *Mathematische Annalen*, 71(1):38–53, 1911. 8

[16] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022. 2, 4, 6

[17] Amir Hertz, Kfir Aberman, and Daniel Cohen-Or. Delta denoising score. In *ICCV*, pages 2328–2337, 2023. 1, 2, 3, 4, 6

[18] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. CLIPScore: a reference-free evaluation metric for image captioning. In *EMNLP*, 2021. 4

[19] Xuan Ju, Ailing Zeng, Yuxuan Bian, Shaoteng Liu, and Qiang Xu. Direct inversion: Boosting diffusion-based editing with 3 lines of code. *arXiv preprint arXiv:2310.01506*, 2023. 2, 6

[20] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *NeurIPS*, 35:26565–26577, 2022. 2

[21] 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 *CVPR*, pages 6007–6017, 2023. 2

[22] Bingxin Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. In *CVPR*, pages 9492–9502, 2024. 2

[23] Jeongsol Kim, Geon Yeong Park, and Jong Chul Ye. Dreamsampler: Unifying diffusion sampling and score distillation for image manipulation. In *ECCV*, pages 398–414. Springer, 2025. 2, 4, 6

[24] Diederik P Kingma, Max Welling, et al. Auto-encoding variational bayes, 2013. 3

[25] Gwanhyeong Koo, Sunjae Yoon, Ji Woo Hong, and Chang D Yoo. Flexiedit: Frequency-aware latent refinement for enhanced non-rigid editing. In *ECCV*, pages 363–379. Springer, 2024. 2, 4, 1

[26] Juil Koo, Chanho Park, and Minhyuk Sung. Posterior distillation sampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13352–13361, 2024. 2

[27] R Krishnamoorthi and Sheba Kezia Malarchelvi. Image adaptive watermarking with visual model in orthogonal polynomials based transformation domain. *International Journal of Signal Processing*, 5(2):146–153, 2009. 2

[28] Gihyun Kwon and Jong Chul Ye. Diffusion-based image translation using disentangled style and content representation. *arXiv preprint arXiv:2209.15264*, 2022. 4, 6

[29] Ninareh Mehrabi, Palash Goyal, Apurv Verma, Jwala Dhamala, Varun Kumar, Qian Hu, Kai-Wei Chang, Richard Zemel, Aram Galstyan, and Rahul Gupta. Resolving ambiguities in text-to-image generative models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14367–14388, 2023. 2

[30] Chenlin Meng, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021. 2

[31] Daiki Miyake, Akihiro Iohara, Yu Saito, and Toshiyuki Tanaka. Negative-prompt inversion: Fast image inversion for editing with text-guided diffusion models. *arXiv preprint arXiv:2305.16807*, 2023. 2

[32] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *CVPR*, pages 6038–6047, 2023. 2

[33] Brian B Moser, Arundhati S Shanbhag, Federico Raue, Stanislav Frolov, Sebastian Palacio, and Andreas Dengel. Diffusion models, image super-resolution, and everything: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 2024. 2

[34] Hyelin Nam, Gihyun Kwon, Geon Yeong Park, and Jong Chul Ye. Contrastive denoising score for text-guided latent diffusion image editing. In *CVPR*, pages 9192–9201, 2024. 2, 3, 4, 6

[35] Lingzhi Pan, Tong Zhang, Bingyuan Chen, Qi Zhou, Wei Ke, Sabine Süsstrunk, and Mathieu Salzmann. Coherent and multi-modality image inpainting via latent space optimization. *arXiv preprint arXiv:2407.08019*, 2024. 2

[36] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. 4

[37] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022. 2, 3

[38] Archiki Prasad, Elias Stengel-Eskin, and Mohit Bansal. Rephrase, augment, reason: Visual grounding of questions for vision-language models. *arXiv preprint arXiv:2310.05861*, 2023. 2

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

[40] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, pages 10684–10695, 2022. 1, 2, 4

[41] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *NeurIPS*, 35:36479–36494, 2022. 2

[42] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *NeurIPS*, 35:25278–25294, 2022. 4

[43] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *CVPR*, pages 1921–1930, 2023. 4

[44] Jianyi Wang, Zongsheng Yue, Shangchen Zhou, Kelvin CK Chan, and Chen Change Loy. Exploiting diffusion prior for real-world image super-resolution. *IJCV*, pages 1–21, 2024. 2

[45] Wei Wu, Qingnan Fan, Shuai Qin, Hong Gu, Ruoyu Zhao, and Antoni B Chan. Freediff: Progressive frequency truncation for image editing with diffusion models. In *ECCV*, pages 194–209. Springer, 2024. 2, 4

[46] Shaoan Xie, Zhifei Zhang, Zhe Lin, Tobias Hinz, and Kun Zhang. Smartbrush: Text and shape guided object inpainting with diffusion model. In *CVPR*, pages 22428–22437, 2023. 2

[47] Chenyu Yang, Wanrong He, Yingqing Xu, and Yang Gao. Elegant: Exquisite and locally editable gan for makeup transfer. In *ECCV*, pages 737–754. Springer, 2022. 2

[48] Yi Yang, Zhengwei Su, and L Sun. Medical image enhancement algorithm based on wavelet transform. *Electronics letters*, 46(2):120–121, 2010.

[49] Hu Yu, Naishan Zheng, Man Zhou, Jie Huang, Zeyu Xiao, and Feng Zhao. Frequency and spatial dual guidance for image dehazing. In *ECCV*, pages 181–198. Springer, 2022. 2

[50] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, pages 3836–3847, 2023. 2

[51] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, pages 586–595, 2018. 4