HIDE: Detecting Diffusion-Based Inpainting via Latent h-Space Representation

Supplementary Material

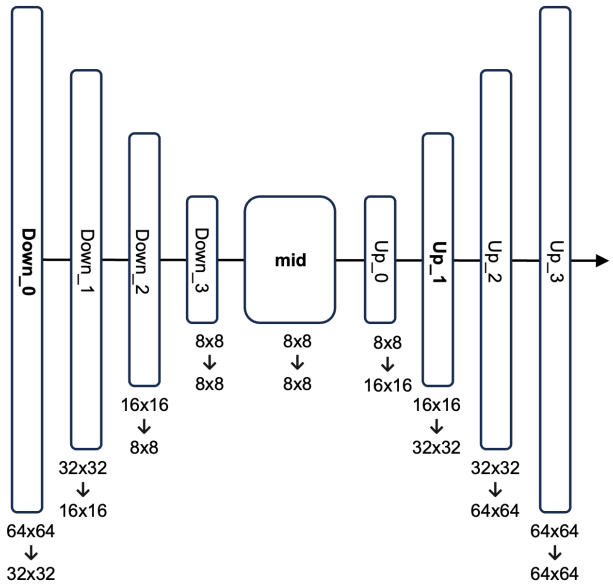

Figure 4. U-Net architecture of Stable Diffusion v1.5. The three evaluated injection locations for h-space features-Down_0, Mid, and Up_1-are annotated. Feature resolutions at each stage are indicated below the blocks.

1. Stable Diffusion v1.5 U-Net Architecture

The U-Net backbone of Stable Diffusion v1.5 [18] consists of four downsampling blocks (Down_0 through Down_3), a single mid-block (Mid), and four upsampling blocks (Up_0 through Up_3), as illustrated in Figure 4. Each block progressively reduces or restores spatial resolution, with feature maps ranging from 64×64 at the shallowest encoder stage to 8×8 at the bottleneck. The model was initialized from the SD v1.2 checkpoint and fine-tuned for 595k steps at 512×512 resolution on the LAION-Aesthetics v2 5+ subset of LAION-5B, a large-scale dataset of over five billion image-text pairs curated for aesthetic quality. For the inpainting variant, the U-Net was further fine-tuned for 440k steps with synthetic masks on the same dataset, with five additional input channels introduced to encode the masked image and the mask itself.

In our experiments, we use this pre-trained U-Net in a frozen state solely as a feature extractor. We evaluate three candidate h-space extraction points that span different stages of the network: the first downsampling block (Down_0), the mid-block bottleneck (Mid), and the first upsampling block (Up_1). As feature resolutions and channel dimensions differ across these stages, a lightweight adapter is applied for Down_0 and Up_1 to unify their representations before injection into the segmentation network via cross-attention.

Algorithm 1 Patchwise Self-Similarity Score

Require: Masked region crop $\mathbf{I} \in \mathbb{R}^{H\times W}$, patch size p, stride s
Ensure: Self-similarity score $\bar{d} \in \mathbb{R}$
1: Extract overlapping patches $\mathcal{P} = \{\mathbf{p}_1, \ldots, \mathbf{p}_N\}$
2: **for** each $\mathbf{p}_i \in \mathcal{P}$ **do**
3: $\quad \hat{\mathbf{p}}_i \leftarrow \mathbf{p}_i / \|\mathbf{p}_i\|_2$
4: $\quad d_i \leftarrow \min_{j\neq i} \mathrm{CosDist}(\hat{\mathbf{p}}_i, \hat{\mathbf{p}}_j)$
5: **end for**
6: **return** $\bar{d} \leftarrow \frac{1}{N}\sum_i d_i$ ▷ lower $\bar{d} \Rightarrow$ more redundant

2. Local Consistency as a Detection Signal

GAN-based inpainting is known to introduce local discontinuities such as checkerboard artifacts due to transposed convolutions, which manifest as structural discordance detectable within small regions [16, 25]. Diffusion-based inpainting, by contrast, generates content through iterative denoising conditioned on global image context, a fundamentally different generative mechanism. We hypothesize that this process produces the opposite effect: rather than local inconsistency, manipulated regions may exhibit abnormally high structural uniformity, as the iterative conditioning tends to over-regularize local textures.

To test this hypothesis, we propose a patchwise self-similarity metric, formally described in Algorithm 1. Overlapping patches of fixed size and stride are extracted from the masked region, and for each patch, the minimum cosine distance to its nearest patch neighbor is computed as a proxy for structural redundancy. A lower minimum distance indicates that the patch has a near-duplicate elsewhere in the region, reflecting unnaturally high local consistency. Figure 5 illustrates this effect: patches extracted from diffusion-inpainted regions visibly exhibit higher structural redundancy compared to those from real regions, with more patches sharing similar local structure across the inpainted area. Averaging these scores across all patches yields a single self-similarity score for the region.

Across 750 image pairs, real regions yield an average score of 0.0473 versus 0.0297 for diffusion-inpainted regions, confirming our hypothesis that manipulated regions exhibit abnormally high local structural consistency. This quantitatively supports the phenomenon of semantic oversmoothing introduced in Section 1 of the main paper, and

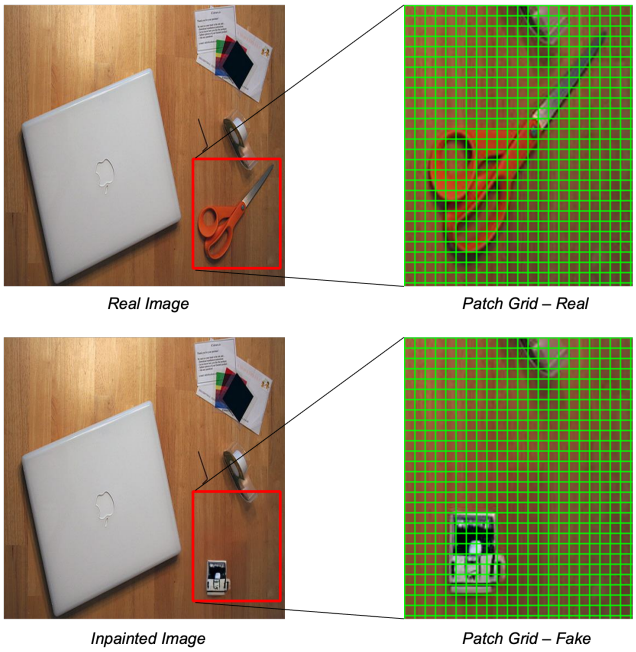

Figure 5. Visualization of patchwise self-similarity extraction. The left column shows real and inpainting examples, with red boxes indicating the manipulated region. The right column shows magnified views of the patch grids overlaid on each region. Patches from the fake region exhibit higher structural redundancy, which translates to lower self-similarity distances.

Auxiliary Feature	None	FT	Gabor	Self-Sim.
AUC	0.9772	0.9796	**0.9824**	0.9811
PR AUC	0.8128	0.8345	**0.8454**	0.8341

Table 7. Ablation on auxiliary feature types. Gabor-based input yields the best performance among tested variants.

demonstrates that self-similarity serves as a complementary detection signal alongside frequency-domain features.

As shown in Table 7, incorporating self-similarity as an auxiliary feature yields competitive performance, outperforming both the baseline and FT-based features, and falling only slightly behind Gabor-based features in both AUC and PR AUC. This confirms that structural redundancy provides meaningful localization cues, while also motivating our final choice of Gabor filters as the primary auxiliary input given their superior directional sensitivity at inpainting boundaries.