HalluGen: Synthesizing Realistic and Controllable Hallucinations for Evaluating Image Restoration

Supplementary Material

Table of Contents

- **6. HalluGen**
  - Sec. 6.1 Implementation Details
  - Sec. 6.2 Ablations
  - Sec. 6.3 Qualitative Results
- **7. Dataset**
  - Sec. 7.1 Instructions for Dataset
  - Sec. 7.2 Computational Cost
  - Sec. 7.3 Limitations
  - Sec. 7.4 License and Ethics Statement
- **8. SHAFE**
  - Sec. 8.1 Implementation Details
  - Sec. 8.2 Ablations
  - Sec. 8.3 Localization Results
  - Sec. 8.4 Limitations of SHAFE
  - Sec. 8.5 Inference Speed
- **9. Hallucination Detectors**
- **Figures**
  - Fig. 9: Effect of temperature τ in softmax aggregation for SHAFE (AUC vs. τ)
  - Fig. 10: Visual results of SHAFE heatmap on both HalluGen and real restoration outputs
  - Fig. 11: Visual examples of SHAFE limitations
  - Fig. 12: Inference time comparison for baseline metrics vs. SHAFE-ResNet50
  - Fig. 13: Network archtiecture of reference-free hallucination detector
  - Fig. 14: Visual results of HalluGen on brain MRI
  - Fig. 15: Visual results of HalluGen on MVTec AD
  - Fig. 16: Visual results of HalluGen on ImageNet
- **Tables**
  - Tab. 5: HalluGen ablations using mean squared error within masked region
  - Tab. 6: SHAFE ablations using AUC on hallucination detection
  - Tab. 7: Impact of feature-layer selection on SHAFE hallucination detection

6. HalluGen

6.1. Implementation Details

We provide additional implementation details for HalluGen. Intrinsic and extrinsic generation procedures are summarized in Algo. 1 and Algo. 2. We first obtain a non-hallucinated baseline x_{base} using DPS [10], then sample candidate patches for hallucination injection. Patches failing entropy or background thresholds are rejected and resampled. HalluGen is initialized either from Gaussian noise (when $t_{\text{skip}} = \text{None}$) or from a partially noised version of x_{base} at timestep t_{skip}.

At each reverse step, we compute the Tweedie estimate $\hat{x}_0(x_t)$. When interpolation is enabled, $\hat{x}_0(x_t)$ is blended with x_{base} outside the mask to preserve fidelity in non-masked regions. Intrinsic generation modifies only the measurement-consistency term, whereas extrinsic additionally applies a feature-space loss.

To maintain realism and avoid patch artifacts, all ascent weights are set to zero after timestep t_{stop}, following LocalDiffusion [22], allowing the diffusion prior to refine boundaries in later steps. Finally, the Hallucination Verification Module (HVM) checks whether masked deviations satisfy the taxonomy criteria (Eqs. 2, 3) via Cohen's d; samples failing the threshold τ_{hvm} are rejected and regenerated.

Hyperparameters. We use the following default settings:

- Diffusion solver: DDPM [17]
- Diffusion parameterization: ε_θ
- Diffusion steps: $T = 1000$
- Skip timestep: $t_{\text{skip}} = 200$
- Measurement-consistency weight: $\lambda_t = 1.0$
- Intrinsic ascent weight: $\gamma_t = 0.002$
- Extrinsic ascent weights: $\gamma_{1,t}, \gamma_{2,t} = 0.007$
- Feature-loss encoder: MedSAM [9]
- Entropy / background thresholds: $1.4/0.05$
- HVM threshold: $\tau_{\text{hvm}} = 0.1$
- Patch size: 16–24
- Number of patches: 1–3
- Ascent cut-off timestep: $t_{\text{stop}} = 10$

6.2. Ablations

In this section, we ablate each HalluGen component: HVM, entropy-based patch selection, and the feature-space loss.

HVM maximizes hallucination while adhering to hallucination taxonomy. HVM enforces the intrinsic/extrinsic definitions in Equations 2 and 3 by rejecting samples that do not exhibit sufficient deviation in masked regions. As shown in Tab. 5, enabling HVM increases measurement- and image-space losses for intrinsic cases by roughly 7% and 20%, indicating stronger violations of data fidelity as required. For extrinsic cases, HVM lowers measurement-space error while increasing image-space deviation by over 56%, enforcing larger semantic changes while more strictly preserving measurement consistency. These results confirm that HVM improves taxonomy compliance by pushing intrinsic hallucinations toward greater measurement inconsis-

Algorithm 1 HalluGen – Intrinsic Hallucination

Require: Ground truth x_{gt}, measurement y, forward operator $\mathcal{A}(\cdot)$
Require: Diffusion model μ_θ, step size $\{\lambda_t\}$, $\{\gamma_t\}$
Require: Non-hallucinated baseline x_{base}
Require: Optional skip timestep t_{skip} (or None), ascent cutoff timestep t_{stop}
Require: Interpolation flag interp $\in \{\text{true}, \text{false}\}$, interpolation schedule w_t

Sample hallucination mask $m \in \{0,1\}^{H\times W}$ by entropy-based patch selection
if t_{skip} is None **then**
  $t_{\text{start}} \leftarrow T$
  Sample $x_T \sim \mathcal{N}(0, I)$
else
  $t_{\text{start}} \leftarrow t_{\text{skip}}$
  Sample $x_{t_{\text{start}}} \sim q(x_{t_{\text{start}}} \mid x_{\text{base}})$ ▷ forward process x_{base}
end if
for $t = t_{\text{start}}, t_{\text{start}} - 1, \ldots, 1$ **do**
  Compute Tweedie estimate $\hat{x}_0(x_t)$
  if interp = true **then**
    $\hat{x}_0(x_t) \leftarrow (1 - m) \odot \left[w_t\, \hat{x}_0(x_t) + (1 - w_t)\, x_{\text{base}}\right] + m \odot \hat{x}_0(x_t)$
  end if
  $x_{t-1} \leftarrow \mu_\theta(x_t, t) - \lambda_t \nabla_{x_t} \left\|(1 - m) \odot \left(y - \mathcal{A}(\hat{x}_0)\right)\right\|_2^2$
  if $t > t_{\text{stop}}$ **then**
    $\gamma_t^{\text{eff}} \leftarrow \gamma_t$
  else
    $\gamma_t^{\text{eff}} \leftarrow 0$ ▷ turn off ascent near the end
  end if
  $x_{t-1} \leftarrow x_{t-1} + \gamma_t^{\text{eff}} \nabla_{x_t} \left\|m \odot \left(y - \mathcal{A}(\hat{x}_0)\right)\right\|_2^2 + \sigma_t \varepsilon$
end for
$\hat{x} \leftarrow \hat{x}_0(x_0)$
Compute d_{meas} using Eq. 8; if taxonomy constraints are not satisfied, resample m and repeat
return intrinsic hallucinated image $\hat{x}$ and mask m

Algorithm 2 HalluGen – Extrinsic Hallucination

Require: Ground truth x_{gt}, measurement y, forward operator $\mathcal{A}(\cdot)$
Require: Diffusion model μ_θ, feature extractor $F(\cdot)$
Require: Step size $\{\lambda_t\}$, $\{\gamma_{1,t}\}$, $\{\gamma_{2,t}\}$
Require: Non-hallucinated baseline x_{base}
Require: Optional skip timestep t_{skip} (or None), ascent cutoff timestep t_{stop}
Require: Interpolation flag interp $\in \{\text{true}, \text{false}\}$, interpolation schedule w_t

Sample hallucination mask $m \in \{0,1\}^{H\times W}$ by entropy-based patch selection
if t_{skip} is None **then**
  $t_{\text{start}} \leftarrow T$
  Sample $x_T \sim \mathcal{N}(0, I)$
else
  $t_{\text{start}} \leftarrow t_{\text{skip}}$
  Sample $x_{t_{\text{start}}} \sim q(x_{t_{\text{start}}} \mid x_{\text{base}})$ ▷ forward process x_{base}
end if
for $t = t_{\text{start}}, t_{\text{start}} - 1, \ldots, 1$ **do**
  Compute Tweedie estimate $\hat{x}_0(x_t)$
  if interp = true **then**
    $\hat{x}_0(x_t) \leftarrow (1 - m) \odot \left[w_t\, \hat{x}_0(x_t) + (1 - w_t)\, x_{\text{base}}\right] + m \odot \hat{x}_0(x_t)$
  end if
  $x_{t-1} \leftarrow \mu_\theta(x_t, t) - \lambda_t \nabla_{x_t} \left\|y - \mathcal{A}(\hat{x}_0)\right\|_2^2$
  if $t > t_{\text{stop}}$ **then**
    $\gamma_{1,t}^{\text{eff}} \leftarrow \gamma_{1,t}, \quad \gamma_{2,t}^{\text{eff}} \leftarrow \gamma_{2,t}$
  else
    $\gamma_{1,t}^{\text{eff}} \leftarrow 0, \quad \gamma_{2,t}^{\text{eff}} \leftarrow 0$ ▷ turn off ascent near the end
  end if
  $x_{t-1} \leftarrow x_{t-1} + \gamma_{1,t}^{\text{eff}} \nabla_{x_t} \left\|m \odot (\hat{x}_0 - x_{\text{gt}})\right\|_2^2$
  $x_{t-1} \leftarrow x_{t-1} + \gamma_{2,t}^{\text{eff}} \nabla_{x_t} \left\|m \odot (F(\hat{x}_0) - F(x_{\text{gt}}))\right\|_2^2 + \sigma_t \varepsilon$
end for
$\hat{x} \leftarrow \hat{x}_0(x_0)$
Compute d_{meas} and d_{img} using Eq. 9; if taxonomy constraints are not satisfied, resample m and repeat
return extrinsic hallucinated image $\hat{x}$ and mask m

tency and extrinsic hallucinations toward targeted semantic deviation under measurement constraints.

Entropy-based patch selection and feature loss increase semantic deviations. The entropy-based patch selection module and the feature loss are designed to target semantically informative, hallucination-prone regions and to ensure that extrinsic hallucinations alter higher-level semantic content rather than only low-level pixel intensities. In Tab.5, removing entropy-based selection reduces both measurement- and image-space deviations for intrinsic cases, while enabling it increases these deviations by about 70%, producing more localized and meaningful perturbations. For extrinsic cases, omitting the feature loss yields small deviations within the masked region. Including it more than doubles the image-space deviation while only slightly raising measurement-space error, which remains orders of magnitude lower than intrinsic cases. This shows that the feature loss strengthens higher-level semantic modification without compromising the strict measurement consistency required for extrinsic hallucinations.

Table 5. **Effects of HVM, Entropy-based selection and Feature loss on hallucination taxonomy compliance using mean squared error within masked region (N=250).**

Case	HVM	Ent.	Feat.	Meas.	Image
Intrinsic	x	x	x	1.5×10^{-3}	3.8×10^{-3}
Intrinsic	x	v	x	2.5×10^{-3}	6.5×10^{-3}
Intrinsic	v	v	x	3.0×10^{-3}	7.0×10^{-3}
Extrinsic	x	v	x	1.6×10^{-5}	1.5×10^{-3}
Extrinsic	x	x	v	1.7×10^{-5}	2.7×10^{-3}
Extrinsic	x	v	v	2.6×10^{-5}	3.0×10^{-3}
Extrinsic	v	v	v	2.2×10^{-5}	4.7×10^{-3}

6.3. Qualitative Results

We provide additional visual examples in Fig. 14, Fig. 15, and Fig. 16, corresponding to MRI, industrial, and natural images, respectively. These results demonstrate that HalluGen generalizes across diverse imaging domains without modification.

6.4. Failure Case Analysis

We observed three recurring challenges during dataset construction with HalluGen.

(1) Sensitivity to regional texture and homogeneity. Hallucination severity depends on the local structure of the selected region. When masks cover smooth or homogeneous areas, the diffusion prior can dominate and suppress the gradient-ascent signal needed to induce hallucination. For extrinsic cases, such regions also shrink the effective null space of the forward operator $\mathcal{A}(\cdot)$, making semantic deviations harder to produce. These issues are largely mitigated by HVM and entropy-based patch selection, which favor more informative regions.

(2) Limited control over precise morphological changes. Because HalluGen relies on stochastic sampling, it cannot deterministically specify the exact morphology or topology of hallucinated structures. While HalluGen provides spatial control and severity modulation, it does not guarantee identical hallucination patterns across runs, suggesting future work on structure-aware hallucination priors.

(3) Diffusion prior strongly influences hallucination realism. Hallucination realism depends on the data distribution of the diffusion prior. To study this, we trained two MVTec AD diffusion models: (i) class-specific and (ii) multi-class. With identical hyperparameters, the class-specific prior produced more coherent hallucinations, whereas the multi-class prior often generated implausible artifacts. Strong gradients can push samples across object manifolds when the prior spans diverse categories. Lower gradient strengths help stabilize results, but adaptive or class-aware gradient schedules remain an open direction.

7. Dataset

7.1. Instructions for Dataset

The dataset is constructed for low-field MRI enhancement. We first pre-train a diffusion model using axial slices from 1,000 HCP scans and simulate low-field MRI using the forward model from DynamicDPS [23]:

$$\mathcal{A}(x) = \text{Blur}\left(\text{DS}_k\left(\Gamma_\gamma(x)\right)\right), \qquad k = 4,\ \gamma = 0.7, \quad (12)$$

where Γ_γ is a gamma transform, DS_k is k-fold downsampling, and Blur applies spatial smoothing. These parameters create a moderately ill-posed setting—severe degradation leads to uncontrolled global hallucinations, while mild degradation limits the null space needed for extrinsic cases.

We then generate 1,450 non-hallucinated predictions using DPS [10]. These serve as (i) the baseline for timestep skipping and (ii) the reference for interpolation outside the hallucination mask, ensuring hallucinations remain localized. Applying HalluGen with intrinsic and extrinsic settings produces 1,450 samples for each category, resulting in a dataset of 4,350 images in total.

The dataset consists of following: For each ground truth image x_{gt}:

- file ID and slice index for ground truth
- measurement image
- non-hallucinated baseline from DPS [10]
- intrinsic and extrinsic hallucinations from HalluGen
- binary hallucination mask for each type

All data are stored as NumPy arrays (.npy).

7.2. Computational Cost

We used $1\times$ A6000 (48GB VRAM) GPU to generate the dataset. It takes approximately $60s$ to generate intrinsic and $90s$ to generate extrinsic sample with batch size 1. However, since HVM may reject the sample with insufficient hallucination, it can take slightly longer.

7.3. Limitations

The current dataset includes only healthy adult brain MRI scans, as the diffusion prior was trained on the HCP [41]. As a result, the hallucination characteristics may not fully reflect those that occur in pathological or clinically diverse populations. Nonetheless, this release serves as an initial foundation for hallucination analysis in image restoration, providing the first controlled dataset of intrinsic and extrinsic hallucinations.

7.4. License and Ethics Statement

This work utilizes fully synthetic human brain MRI data for dataset construction. The underlying pre-trained diffusion

model is trained on human brain MRI data from the Human Connectome Project (HCP) [41]. Our released dataset consists exclusively of synthetic ground-truth MR images generated by the diffusion model, along with hallucinated outputs produced using HalluGen. We do not redistribute any original HCP imaging data.

All dataset components, including synthetic ground-truth images, non-hallucinated reconstructions, hallucination masks, and hallucinated samples, are released under the CC BY-NC 4.0 International License[1], which permits non-commercial use with appropriate attribution. The source code is released under the MIT License[2].

This dataset is intended strictly for research purposes. It must not be used for clinical decision-making or deployed in real-world medical settings. No additional human subjects were recruited or scanned for this study.

8. SHAFE

8.1. Implementation Details

We provide additional implementation details for SHAFE, illustrated in Algo. 3. SHAFE is a full-reference metric that takes as input the predicted reconstruction $\hat{x}$ and the ground-truth image x_{gt}. Both images are first passed through a low-pass filter to suppress high-frequency artifacts that do not correspond to hallucinations (e.g., checkerboard patterns), thereby reducing false positives.

Feature representations are extracted using a pre-trained encoder (e.g., DINOv3). We use multi shallow-layer features to avoid semantic bias from deeper layers and to emphasize structural and textural cues that better reflect hallucination-related deviations. Cosine distances are computed patch-wise between the feature maps of $\hat{x}$ and x_{gt}.

Traditional image-quality metrics that aggregate patch scores uniformly (e.g., SSIM, LPIPS), but this is suboptimal for hallucination assessment, as hallucinations tend to be sparse and highly localized (see Fig. 10). To address this, inspired by soft-attention pooling [49], we apply a weighted softmax aggregation over patch-level distances, enabling SHAFE to upweight semantically abnormal regions while downweighting benign variations.

Hyperparamters. We use the following settings:

- Low-pass filter radius: 50
- Feature backbone: DINOv3 [40], ResNet50 [51], MedSAM [9]
- Feature layer indices: [1, 2, 3], [1, 2], [-1]
- tau : 0.01, 0.01, 0.005

[1] https://creativecommons.org/licenses/by-nc/4.0/

[2] https://opensource.org/licenses/MIT

Algorithm 3 SHAFE: Semantic Hallucination Assessment via Feature Evaluation

Require: Prediction $\hat{x}$, ground truth x_{gt}
Require: Low-pass filter $\text{LP}(\cdot)$, pretrained encoder $\{f_\ell\}_{\ell\in\mathcal{L}}$ (e.g., DINOv3)
Require: Temperature $\tau > 0$

$\tilde{x} \leftarrow \text{LP}(\hat{x}), \quad \tilde{x}_{\text{gt}} \leftarrow \text{LP}(x_{\text{gt}})$
for each layer $\ell \in \mathcal{L}$ **do**
  $F_\ell \leftarrow f_\ell(\tilde{x}), \quad G_\ell \leftarrow f_\ell(\tilde{x}_{\text{gt}})$
end for
$F \leftarrow \text{Concat}\big(\{F_\ell\}_{\ell\in\mathcal{L}}\big), \quad G \leftarrow \text{Concat}\big(\{G_\ell\}_{\ell\in\mathcal{L}}\big)$
Reshape F, G into patch features $\{F_i\}_{i=1}^N$, $\{G_i\}_{i=1}^N$
for $i = 1$ to N **do**
  $\delta_{\cos,i} \leftarrow 1 - \dfrac{\langle F_i, G_i\rangle}{\|F_i\|_2\,\|G_i\|_2}$ ▷ cosine distance per patch
end for
$w_i \leftarrow \dfrac{\exp(\delta_{\cos,i}/\tau)}{\sum_{j=1}^N \exp(\delta_{\cos,j}/\tau)}$ for $i = 1, \ldots, N$
$\text{SHAFE} \leftarrow \sum_{i=1}^N w_i\,\delta_{\cos,i}$
return SHAFE ▷ optionally also return patch map $\{w_i\delta_{\cos,i}\}$

8.2. Ablations

We ablate the components of SHAFE, including the low-pass (LP) filter, weighted softmax aggregation, temperature τ, and feature-layer selection. All experiments use a subset of our HalluGen dataset (N=300) for both intrinsic and extrinsic hallucinations.

Effect of Low-Pass Filtering and Weighted-softmax Aggregation. Table 6 shows that both LP filtering and weighted softmax aggregation improve hallucination detection (AUC). LP filtering suppresses high-frequency noise, helping SHAFE emphasize coherent structural differences. Weighted softmax aggregation further highlights patches with stronger feature deviations, and already outperforms uniform pooling, reflecting the spatially localized nature of hallucinations. Combining both modules increases AUC from 0.52 to 0.78, demonstrating the complementary benefits of frequency smoothing and adaptive spatial weighting.

Effect of Temperature τ. Figure 9 illustrates the impact of varying the temperature τ in the softmax aggregation. Very small values (e.g., $\tau \leq 0.001$) approximate max-pooling and already produce strong AUC, indicating that hallucinations often manifest as localized feature deviations. Performance remains stable until $\tau \approx 0.02$, after which AUC decreases gradually as the softmax distribution becomes more uniform and less discriminative. Nonethe-

Table 6. **Ablation of low-pass filtering (LP) and weighted-softmax aggregation in SHAFE, evaluated using AUC on hallucination detection (N=300).** Both components improve detection sensitivity, and their combination yields the highest performance, indicating that suppressing high-frequency noise while adaptively weighting salient patches is crucial for detecting hallucinations.

LP	Weighted softmax	AUC
x	x	0.52
v	x	0.55
x	v	0.64
v	v	**0.78**

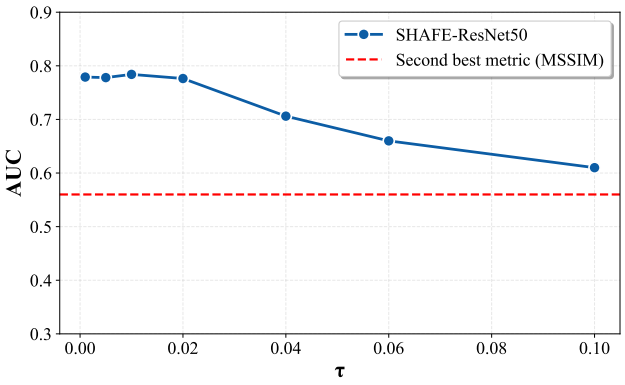

Figure 9. **Effect of temperature τ in softmax aggregation for SHAFE (AUC vs. τ).** Small temperatures approximate max-pooling and yield strong performance, stable until $\tau \approx 0.02$. Larger values smooth patch weighting and gradually reduce discriminability, yet SHAFE remains superior to the second-best metric across all tested values.

less, SHAFE continues to outperform the second-best baseline across all tested values, highlighting robustness to τ.

Impact of Feature-Layer Selection. Table 7 reports the effect of feature-layer selection when using SHAFE-ResNet50. Using only deeper layers degrades performance, likely due to dataset-specific and task-specific bias that do not fully align with medical and industrial domains. Shallow features capture texture and low-level structure, which are more sensitive to fine-grained hallucinations. The best performance is achieved when combining multiple shallow layers (AUC=0.78), supporting our design choice that multi-layer early features provide richer representations for hallucination scoring.

8.3. Localization

Fig. 10 presents the localization performance of SHAFE on both HalluGen samples and real restoration outputs generated by ESRGAN [44] and SwinIR [30]. SHAFE consistently highlights hallucinated regions such as distorted lateral ventricles and disrupted sulcal boundaries, demonstrating its ability to detect both synthetic and naturally occurring failures. The resulting heatmaps are characteristically sparse, emphasizing localized deviations rather than diffuse artifacts. This sparsity further justifies the use of weighted softmax aggregation over uniform pooling, as it enables SHAFE to selectively amplify subtle but semantically meaningful hallucinated features.

Table 7. **Impact of feature-layer selection on SHAFE hallucination detection (AUC, N=300) using SHAFE-ResNet50.** Deep layers alone degrade performance due to semantic bias from ImageNet pretraining, whereas combining shallow layers provides the highest AUC, supporting the use of multi-layer early features for hallucination sensitivity.

Layer 1	Layer 2	Layer 3	AUC
v	x	x	0.75
x	v	x	0.76
x	x	v	0.63
v	v	x	**0.78**
v	v	v	0.72

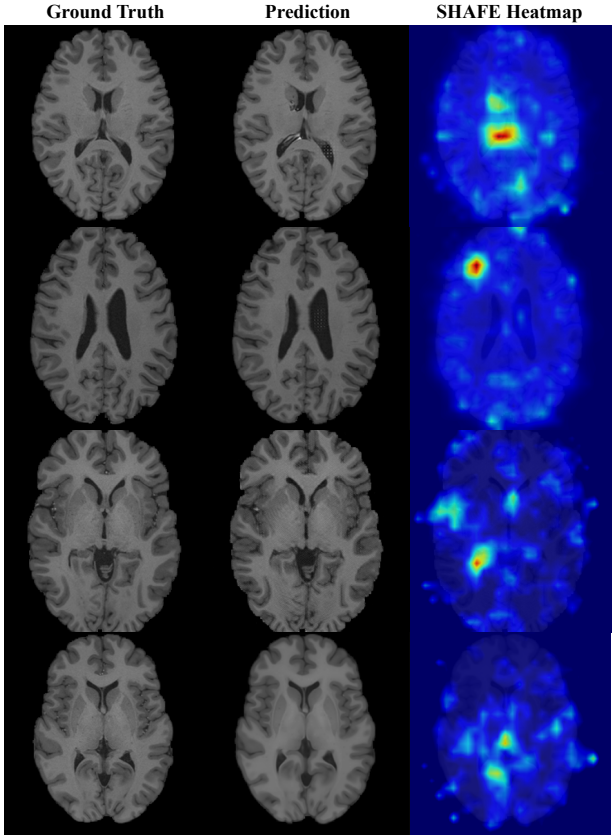

Figure 10. **Visual results of SHAFE heatmap on both synthetic (HalluGen) and real restoration outputs.** First and second rows are extrinsic and intrinsic examples from HalluGen and third and fourth are real restoration outputs from ESRGAN and SwinIR.

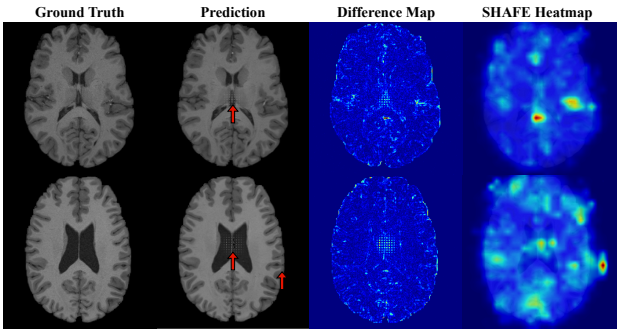

Figure 11. **Visual examples of SHAFE limitations (red arrows).** SHAFE is less sensitive to high-frequency reconstruction artifacts (e.g., periodic grid patterns) due to the use of low-pass filtering, which suppresses non-semantic noise. In addition, SHAFE may highlight boundary regions where intensity differences are large but carry limited semantic relevance.

8.4. Limitations of SHAFE

Fig. 11 illustrates representative limitations of SHAFE. First, SHAFE is less sensitive to high-frequency artifacts (e.g., periodic grid patterns) due to the low-pass filtering step, which suppresses high-frequency content because hallucinations in our taxonomy mainly occur in the low- to mid-frequency range. Consequently, images with such artifacts may obtain SHAFE scores similar to clean ones. Second, SHAFE can assign high responses to large intensity differences in semantically uninformative regions, such as brain boundaries, where intensity mismatches are high but structural deviation is minimal.

8.5. Inference Speed

Fig. 12 compares the inference time of SHAFE-ResNet50 with baseline metrics in our hallucination benchmark. While SHAFE is naturally slower than pixel-based metrics such as PSNR and SSIM, it remains significantly more efficient than feature-based alternatives: SHAFE is about 2.4× faster than LPIPS and 9× faster than DISTS. This shows that the combination of low-pass filtering, shallow feature extraction, and weighted softmax aggregation provides strong detection performance without the heavy computational cost of deep feature metrics. All timings were measured on a local machine with an AMD Ryzen 7 CPU.

9. Reference-free Hallucination Detector

We describe the architecture of our reference-free hallucination detector. As shown in Fig. 13, the detector takes the measurement y and predicted reconstruction $\hat{x}$ as inputs, both of which are first low-pass filtered to suppress high-frequency noise. The filtered images are passed through a shallow ResNet-50 feature extractor, and their features are concatenated before a binary classification head. We also

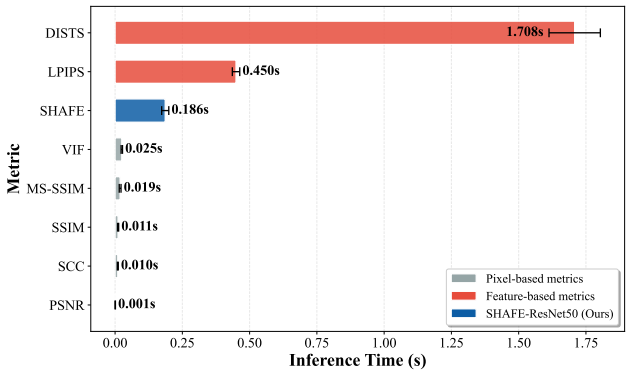

Figure 12. **Inference time comparison for baseline metrics vs. SHAFE-ResNet50.** Despite using feature extraction and weighted softmax, SHAFE remains computationally efficient, running significantly faster than other feature-based metrics such as LPIPS and DISTS. This demonstrates that SHAFE achieves superior detection performance without incurring substantial computational overhead. AMD Ryzen 7 CPU was used for testing.

experimented with concatenating the raw inputs prior to feature extraction, but this caused the model to focus on low-level differences (e.g., small shifts or texture changes) rather than true semantic discrepancies. The detector is trained with cross-entropy loss for 100 epochs using a learning rate of 0.001 on HalluGen-generated samples. Because it relies only on the measurement–prediction pair, the detector supports fully reference-free hallucination detection.

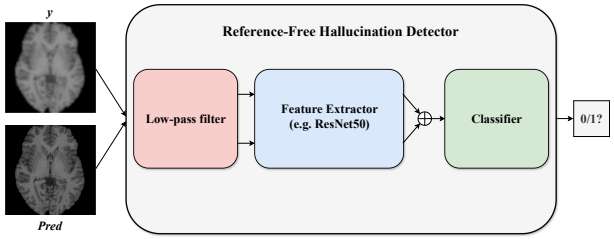

Figure 13. **Network architecture of reference-free hallucination detector.** Prediction and measurement images are fed into the model, applied low-pass filter to remove high-frequency noise and then classified using a simple CNN architecture such as ResNet50.

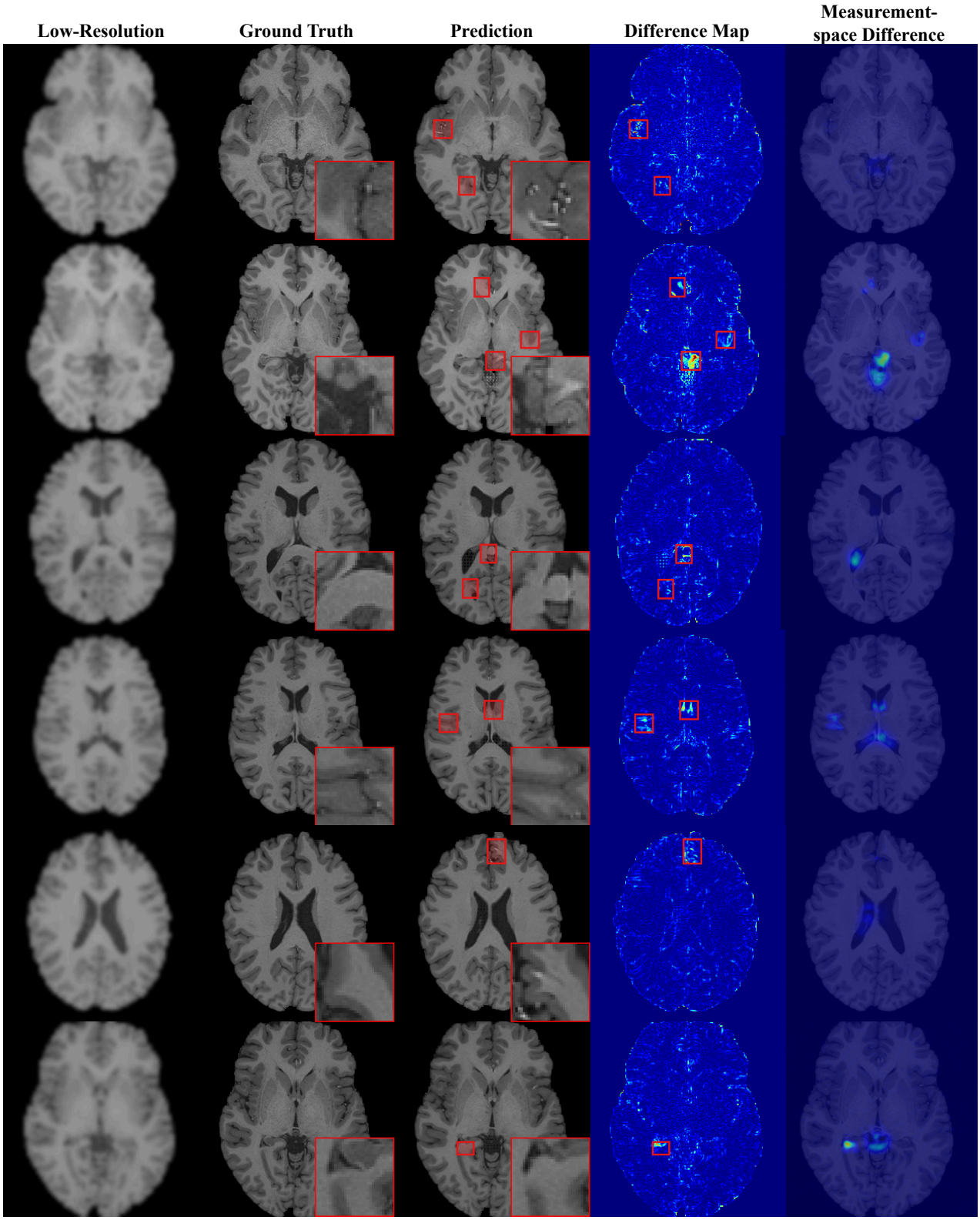

Figure 14. **Visual results of HalluGen examples on brain MRI.** Odd-numbered rows show intrinsic hallucinations, and even-numbered rows show extrinsic hallucinations.

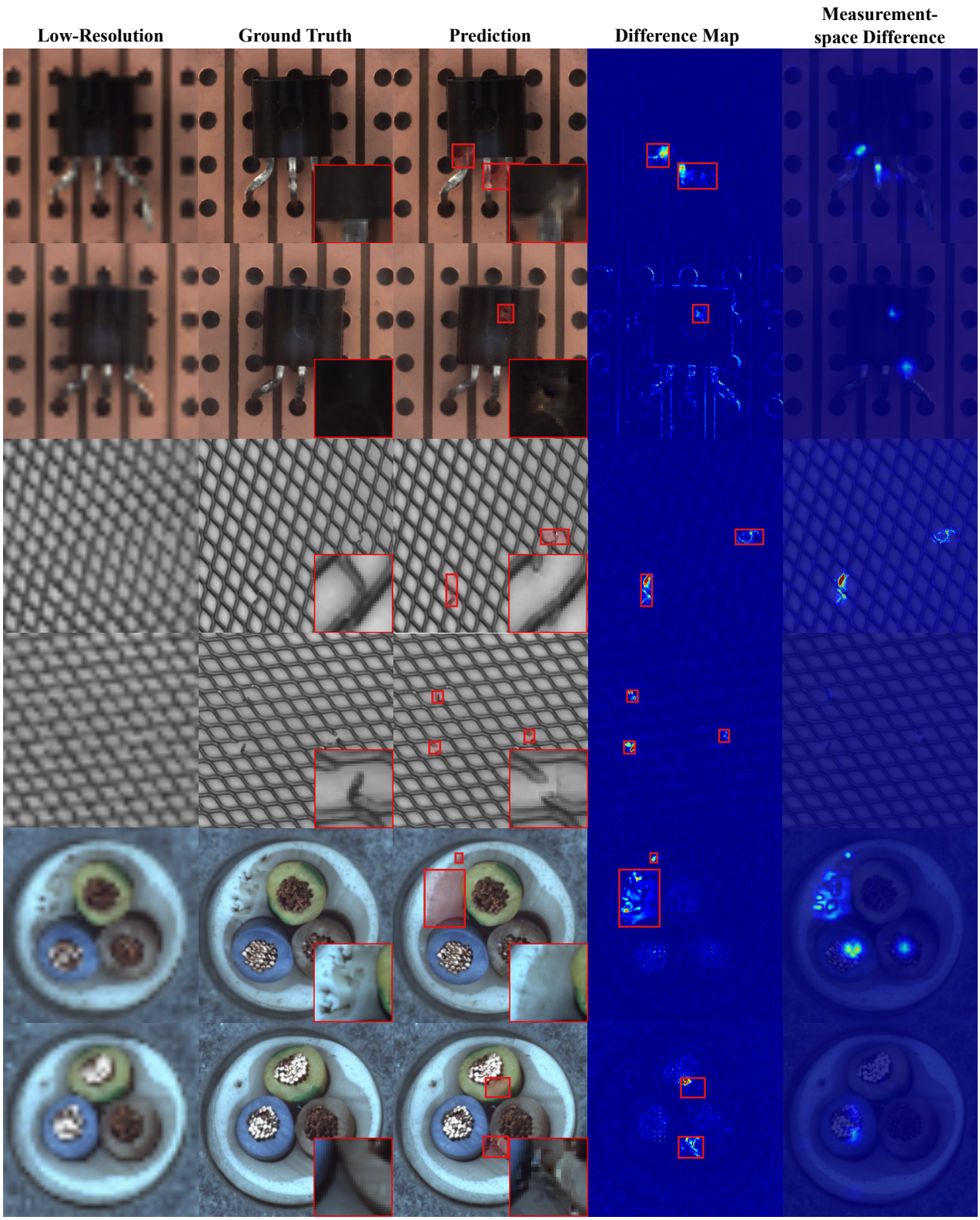

Figure 15. **Visual results of HalluGen examples on MVTec AD.** Odd-numbered rows show extrinsic hallucinations, and even-numbered rows show intrinsic hallucinations. Row 4 and 5 use ground truth mask from MVTec AD to show the flexibility of HalluGen with different size and shape of masks.

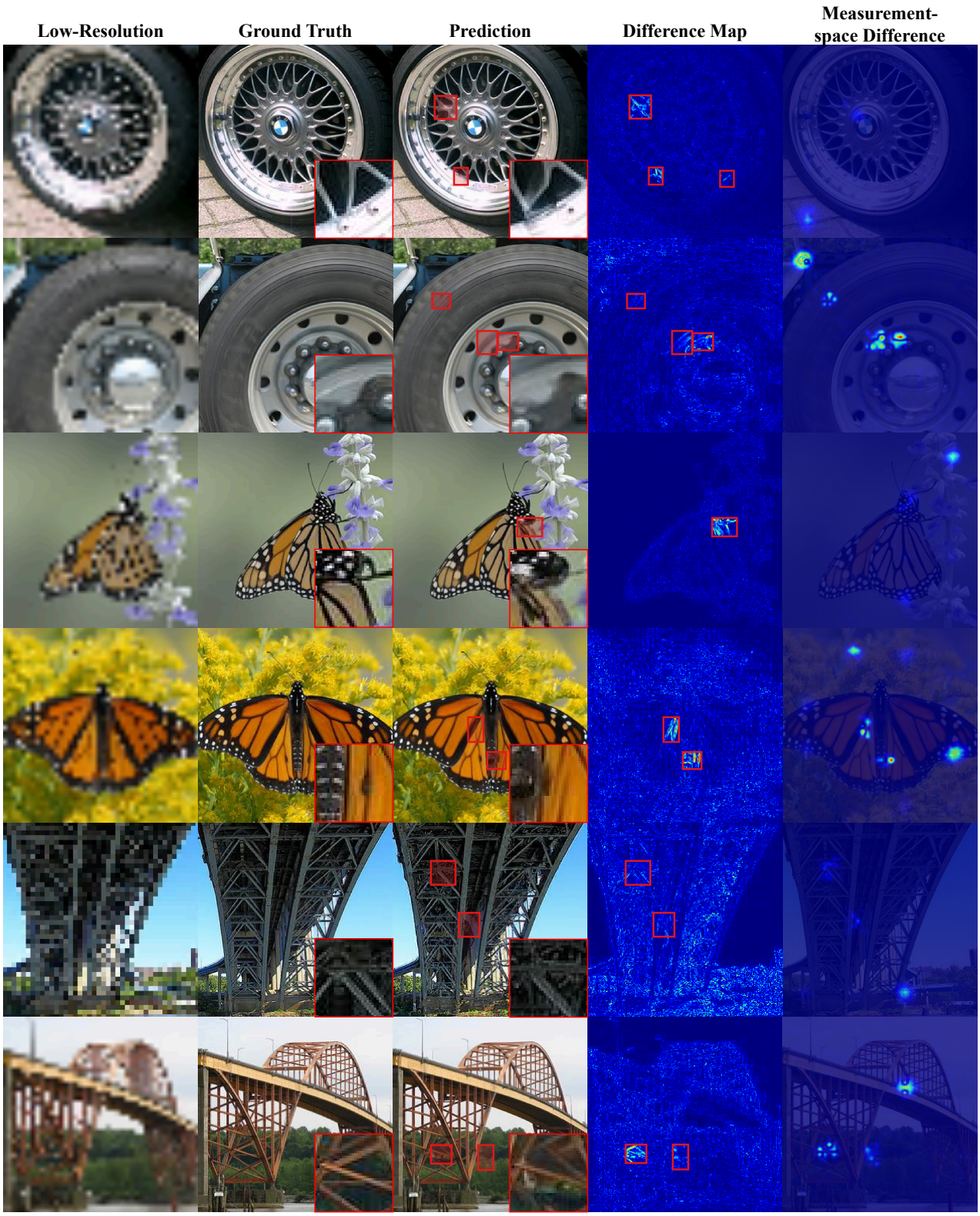

Figure 16. **Visual results of HalluGen examples on ImageNet.** Odd-numbered rows show extrinsic hallucinations, and even-numbered rows show intrinsic hallucinations.