Appendices

A. Derivation

We provide the derivation of Eq. (8), (9), (13) and (15) in the main text. For Eq. (8), we have

$$
\begin{aligned}
\tilde{\boldsymbol{x}}_{0,t}^{h} &= \frac{\boldsymbol{x}_t - \sqrt{1-\bar{\alpha}_t}\left[(1+w)\boldsymbol{\epsilon_\theta}(\boldsymbol{x}_t,t,h) - w\boldsymbol{\epsilon_\theta}(\boldsymbol{x}_t,t,\emptyset)\right]}{\sqrt{\bar{\alpha}_t}} \\
&= \frac{\boldsymbol{x}_t}{\sqrt{\bar{\alpha}_t}} - A_t\left[(1+w)\boldsymbol{\epsilon_\theta}(\boldsymbol{x}_t,t,h) - w\boldsymbol{\epsilon_\theta}(\boldsymbol{x}_t,t,\emptyset)\right] \\
&= \frac{\boldsymbol{x}_t}{\sqrt{\bar{\alpha}_t}} + B_t\left[(1+w)\boldsymbol{s_\theta}(\boldsymbol{x}_t,t,h) - w\boldsymbol{s_\theta}(\boldsymbol{x}_t,t,\emptyset)\right] \\
&= \frac{\boldsymbol{x}_t}{\sqrt{\bar{\alpha}_t}} + B_t\left[(1+w)\nabla_{\boldsymbol{x}_t}\log p_\theta\left(\boldsymbol{x}_t \mid h\right) - w\nabla_{\boldsymbol{x}_t}\log p_\theta\left(\boldsymbol{x}_t\right)\right] \\
&= \frac{\boldsymbol{x}_t}{\sqrt{\bar{\alpha}_t}} + B_t\left[(1+w)\nabla_{\boldsymbol{x}_t}\log p_\theta\left(h \mid \boldsymbol{x}_t\right) + \nabla_{\boldsymbol{x}_t}\log p_\theta\left(\boldsymbol{x}_t\right)\right] \\
&= \frac{\boldsymbol{x}_t}{\sqrt{\bar{\alpha}_t}} + B_t\nabla_{\boldsymbol{x}_t}\log\left[p_\theta(\boldsymbol{x}_t)p_\theta\left(h \mid \boldsymbol{x}_t\right)^{1+w}\right] \\
&= \frac{\boldsymbol{x}_t}{\sqrt{\bar{\alpha}_t}} + B_t\nabla_{\boldsymbol{x}_t}\log\tilde{p}_\theta\left(\boldsymbol{x}_t \mid h\right)
\end{aligned}
$$

where $A_t = \frac{\sqrt{1-\bar{\alpha}_t}}{\sqrt{\bar{\alpha}_t}}$ and $B_t = \frac{1-\bar{\alpha}_t}{\sqrt{\bar{\alpha}_t}}$. The score functions are defined as $\boldsymbol{s_\theta}(\boldsymbol{x}_t,t,h) = \nabla_{\boldsymbol{x}_t}\log p_\theta\left(\boldsymbol{x}_t \mid h\right)$ and $\boldsymbol{s_\theta}(\boldsymbol{x}_t,t,\emptyset) = \nabla_{\boldsymbol{x}_t}\log p_\theta\left(\boldsymbol{x}_t\right)$. For Eq. (9), we substitute $\boldsymbol{x}_t = \sqrt{\bar{\alpha}_t}\boldsymbol{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_t$ into Eq. (8).

For Eq. (13) and (15), we have

$$
\begin{aligned}
M_t &= \frac{\left\|\tilde{\boldsymbol{x}}_{0,t}^{h} - \tilde{\boldsymbol{x}}_{0,t}^{\emptyset}\right\|_2^2}{d} \\
&= \frac{A_t^2(1+w)^2}{d}\left\|\boldsymbol{\epsilon_{\theta,t}^{h}} - \boldsymbol{\epsilon_{\theta,t}^{\emptyset}}\right\|_2^2 \\
&= \frac{B_t^2(1+w)^2}{d}\left\|\boldsymbol{s_{\theta,t}^{h}} - \boldsymbol{s_{\theta,t}^{\emptyset}}\right\|_2^2 \\
&= \frac{C_t^2}{d}\left\|\nabla_{\boldsymbol{x}_t}\log p_\theta\left(\boldsymbol{x}_t \mid h\right) - \nabla_{\boldsymbol{x}_t}\log p_\theta\left(\boldsymbol{x}_t\right)\right\|_2^2 \\
&= \frac{C_t^2}{d}\left\|\nabla_{\boldsymbol{x}_t}\log p_\theta\left(h \mid \boldsymbol{x}_t\right)\right\|_2^2
\end{aligned}
$$

where $C_t = B_t(1+w)$

$$
\begin{aligned}
\boldsymbol{e}_t^h &= \tilde{\boldsymbol{x}}_{0,t}^{h} - \boldsymbol{x}_0 \\
&= B_t\left[(1+w)\nabla_{\boldsymbol{x}_t}\log p_\theta\left(\boldsymbol{x}_t \mid h\right) - w\nabla_{\boldsymbol{x}_t}\log p_\theta\left(\boldsymbol{x}_t\right) - \nabla_{\boldsymbol{x}_t}\log q(\boldsymbol{x}_t)\right] \\
&= B_t\left((1+w)\nabla_{\boldsymbol{x}_t}\log p_\theta\left(h \mid \boldsymbol{x}_t\right) + \Delta\boldsymbol{s}_t\right)
\end{aligned}
$$

B. Data Preprocessing and Training Hyperparameters

We standardized preprocessing for both the BraTS21 and ATLAS v2.0 datasets. Each 3D subject was normalized by dividing it by the 99th percentile intensity of foreground voxels, and pixel values were then scaled to the range of $[-1, 1]$. All samples are interpolated to 128×128.

The backbone U-net is adopted from the previous work [2]. Our model is trained on 2 Nvidia A100 GPUs with 80GB memory. The training hyperparameters are summarized in Tab. 1, and we used the same hyperparameters for both dataset.

Diffusion steps	1000
Noise schedule	linear
Channels	128
Heads	2
Attention resolution	32,16,8
Channel multiplier	1, 1, 2, 3, 4
Dropout	0.1
EMA rate	0.9999
Optimiser	AdamW
Learning rate	$1e^{-4}$
β_1, β_2	0.9, 0.999
Global batch size	64
Null label ratio	0.1
dropout	0.1

Table 1. Training hyperparameters used in our method.

C. Fixed Guidance Selection and Segmentation

We illustrate the fixed guidance selection in Algorithm 1 and outline the complete segmentation process in Algorithm 2.

```
Algorithm 1: Selection of fixed guidance $w^*$
Input: $n$ sorted candidates $[w_1, ..., w_n]$, validation set with image-level labels
for each candidate $w_i$:
    calculate cosine similarity for each sample in validation set with Eq. 18
    classify each samples in validation set with cosine similarity threshold $Cos_{w_i}$
    get the maximal classification accuracy $Acc_{w_i}$ using the optimal threshold $Cos^*_{w_i}$
end for
$w = \arg\max_w Acc_{w_i}$
$w^* = w_i > w$ with $\frac{Acc_{w_i}}{Acc_w} \approx 0.98$
$Cos_{w^*} = Cos^*_{w_i}$ # corresponding threshold
Return $w^*$, $Cos_{w^*}$
```

D. Gradient for Segmentation

We investigate the gradient $\nabla_{\boldsymbol{x}_t} \log p_\theta\left(h|\boldsymbol{x}_t\right)$ as the SAMs for segmentation, while keep other settings unchanged. Here, we use the same BraTS21 testing data, focusing on the unhealthy setup with four configurations: (i) DDIM $\nabla^2 \log p_\theta$: the gradient is directly used as SAMs for segmentation; (ii) DDIM $\nabla^2 \log p_\theta|_{t=t_e}$: only the gradient at the end step is ued for segmentation; (iii) DDIM $C_t^2 \nabla^2 \log p_\theta$: the weighted gradient is used as SAMs; and (iv) DDIM $B_t^2\left((1+w)\nabla \log p_\theta + \Delta \boldsymbol{s}_t\right)^2$: the original SAMs.

The quantitative results are exhibited in Tab. 2. We note that the performance of other configurations is significantly lower compared to using the original SAMs. The last setup **DDIM** $C_t \nabla \log p_\theta$, the weighted gradient, achieved better segmentation results compared to non-weighted gradient SAMs. This is attributed to the weight C_t, a monotonically increasing function. The SAMs with anomalous regions are more weighted. Also, the error term $\Delta \boldsymbol{s}_t$ used in the original SAMs is not considered here, which may alleviate the false detection by the implicit classifier.

```
Algorithm 2: The full segmentation process for a single input $\boldsymbol{x}_0$
Input: fixed guidance $w^*$, input $\boldsymbol{x}_0$
for each time step $t$
    $\tilde{\boldsymbol{x}}^h_{0,t} = \frac{\boldsymbol{x}_t - \sqrt{1-\bar{\alpha}_t}[(1+w^*)\boldsymbol{\epsilon_\theta}(\boldsymbol{x}_t,t,h) - w^*\boldsymbol{\epsilon_\theta}(\boldsymbol{x}_t,t,\emptyset)]}{\sqrt{\bar{\alpha}_t}}$ # Healthy guided prediction Eq. 7
    $\tilde{\boldsymbol{x}}^\emptyset_{0,t} = \frac{\boldsymbol{x}_t - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon_\theta}(\boldsymbol{x}_t,t,\emptyset)}{\sqrt{\bar{\alpha}_t}}$ # unguided prediction Eq. 10
    $M_t = \frac{\left\|\tilde{\boldsymbol{x}}^h_{0,t} - \tilde{\boldsymbol{x}}^\emptyset_{0,t}\right\|_2^2}{d}$ # divergence Eq. 12
    $\boldsymbol{S}_t = \tilde{\boldsymbol{x}}^\emptyset_{0,t} - \boldsymbol{x}_0$ # obtain SAM using Eq. 19
end for
$t_e = \arg\max_t M_t$ # find the end step
$\boldsymbol{H} = \frac{1}{t_e}\sum_{t=1}^{t_e} \boldsymbol{S}_t$ # aggregated SAMs Eq. 20
Obtain the quantile $Q^*$ using Algorithm 1 from the main text
predicted pixel-level labels $= \boldsymbol{H} \geq Q^*$
Return predicted pixel-level labels
```

Methods	DICE	IoU	AUPRC
DDIM $\nabla^2 \log p_\theta$	42.7±0.2	30.6±0.1	42.0±0.0
DDIM $\nabla^2 \log p_\theta\|_{t=t_e}$	53.1±0.0	40.4±0.0	58.8±0.0
DDIM $C_t^2 \nabla^2 \log p_\theta$	57.2±0.1	45.8±0.1	70.3±0.0
***DDIM** $B_t^2\left((1+w)\nabla \log p_\theta + \Delta \boldsymbol{s}_t\right)^2$	**61.5±0.0**	**51.0±0.1**	**75.5±0.1**

Table 2. Segmentation performance on unhealthy samples from BraTS21 dataset using the gradient $\nabla_{\boldsymbol{x}_t} \log p_\theta\left(h|\boldsymbol{x}_t\right)$ as the SAMs. The last setup with $*$ is the original SAMs.

E. More Qualitative Results

We provide more qualitative results for the BraTS21 and ATLAS v2.0 datasets in Fig. 1. It further shows the effectiveness of our method in detecting anomalies and segmenting them. The signal strength of the anomalies is enhanced by the aggregation of SAMs, leading to more accurate segmentation results.

F. Postprocessing for Segmentation

After we obtain the anomaly map, we apply a median filter [1] with kernel size 5 to effectively enhance the performance. Then, we apply the connected component filter to remove the small connected components which is regarded as noise. We apply the same postprocessing to all methods for fair comparison.

G. Discussion and Limitations

All methods showed better results on the BraTS21 dataset than on the ATLAS v2.0 dataset. This disparity arises because DMs are more adept at identifying anomalies that exhibit significant frequency differences, such as tumors on FLAIR MRI, compared to the surrounding healthy tissue. In this case, the difference of healthy and unhealthy distribution is easier to be captured by DMs. In contrast, the ATLAS v2.0 dataset, which consists of T1 MRI, presents more challenging scenarios for anomaly detection due to the subtle frequency differences between healthy and unhealthy regions. During the inference stage, the implicit classifier struggles to accurately capture the anomalous regions, contributing to the less consistent signal of anomalous regions in the SAMs. This inconsistency can lead to the mixing of signals from falsely detected healthy regions, resulting in lower detection accuracy.

Our selection method is specifically designed for the weakly-supervised setting, where unhealthy samples are available for training the guided diffusion model. In unsupervised settings, the unguided diffusion model is typically trained only on healthy samples and evaluated on unhealthy samples. In this scenario, the unguided forward process (UFP) of samples through the diffusion model is not possible, which is a crucial aspect of our method. We leave the exploration of unsupervised settings for future work.

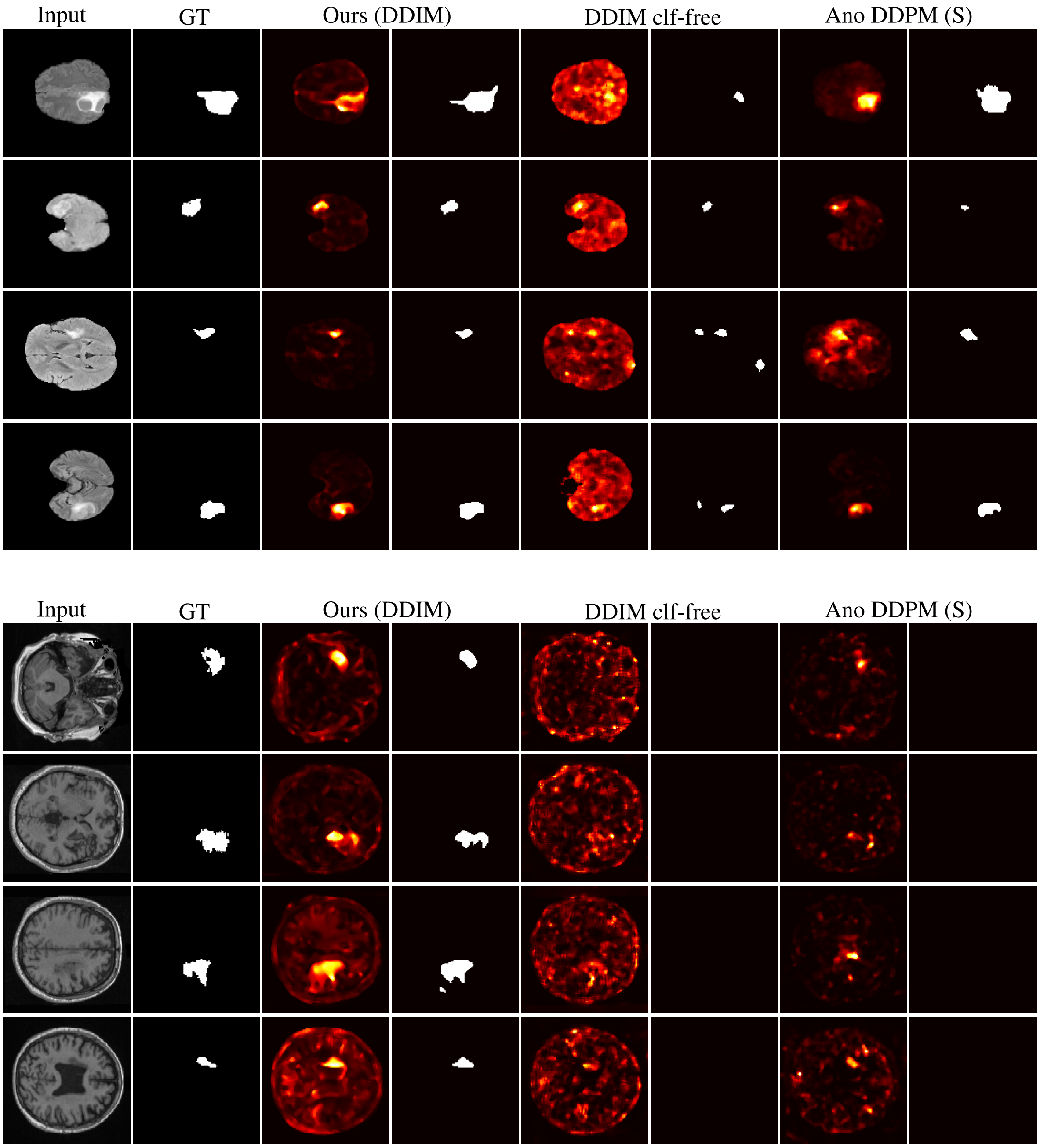

Figure 1. Qualitative Comparison of Anomaly Maps and Segmentation. (a) From the BraTS21 dataset and (b) from the ATLAS v2.0 dataset. The first column displays the original input images, and the second column shows the corresponding ground truth for anomaly segmentation. Subsequent columns present the anomaly maps and segmentation results obtained using our method, AnoFPDM with the DDIM setting, alongside those from the second and third best comparative methods. Each row represents a different sample.

References

[1] Antanas Kascenas, Nicolas Pugeault, and Alison Q O'Neil. Denoising autoencoders for unsupervised anomaly detection in brain mri. In *International Conference on Medical Imaging with Deep Learning*, pages 653–664. PMLR, 2022. 3

[2] Pedro Sanchez, Antanas Kascenas, Xiao Liu, Alison Q O'Neil, and Sotirios A Tsaftaris. What is healthy? generative counterfactual

diffusion for lesion localization. In *MICCAI Workshop on Deep Generative Models*, pages 34–44. Springer, 2022. 2