We assume that the gradients to the input image follow a Gaussian distribution $\nabla_{\boldsymbol{x}_t} L \sim \mathcal{N}(\mathbf{0}, \beta_t \boldsymbol{I})$ with variance β_t. Our target is to denoise the gradients to generate high-quality anomaly-free images. Notice that if the weight decay factor is set to be $\omega = \sqrt{1 - s^2}$, each optimization step becomes a diffusion step with the noise $\epsilon_t \sim \mathcal{N}(\mathbf{0}, s^2 \beta_t \boldsymbol{I})$. We denote the new variance as $\hat{\beta}_t = s^2 \beta_t$. Then we can safely use the diffusion model to denoise the intermediate images during the optimization. Since the variance of noise ϵ_t is relatively small, we denoise the image every N_d optimization step. We demonstrate the sampling process with Algorithm 1.

We visualize the intermediate steps of our gradient denoising process in Fig 7. The anomalous pixels are gradually altered to transform the input image into a normal image. We compare the reconstruction results with other state-of-the-art reconstruction-based anomaly detection methods in Fig. 4. The reconstructed image from our gradient denoising process keeps high-frequency details and successfully removes the anomaly pixels.

4. Experiments

4.1. Dataset and Implementation Details

MVTec-AD We evaluate our proposed method on the MVTec-AD dataset [3], an industrial anomaly detection benchmark that comprises 15 categories, including ten object classes and five texture classes. Each class contains approximately 200 anomaly-free images for training and 100 images with anomalies for testing. The dataset provides pixel-level segmentation ground truth for evaluation. The MVTec-AD dataset contains various anomalies, making it a comprehensive and ideal benchmark for anomaly detection evaluation.

Metrics We assess our pixel-level anomaly segmentation performance with two commonly used threshold-independent metrics: the Area Under the Receiver Operating Characteristic curve (AUROC), and Per-Region-Overlap (PRO) [3]. While AUROC equally measures performance for each pixel, it tends to favor larger area anomalies. To correctly assess the performance on both large and small area anomalies, we also evaluate our method with the PRO. To compute PRO, the area coverage ratios of each connected component are averaged for the same false positive rate. By repeatedly computing the values for the false positive rate from 0 to 0.3, we get a curve, and the normalized integral of this curve is the PRO-score. Unlike AUROC, the PRO metric equally measures the performance for large and small anomalies, which makes it a balanced evaluation metric for industrial anomaly detection.

Implementation details We train our diffusion model separately for each category of MVTec-AD[3]. We adopt the UNet network design with attention modules from improved diffusion [23]. Please refer to the supplementary for network details. The timestep for the diffusion process is set to be 1000 for training and 250 for reverse sampling. The diffusion model is trained for 10,000 iterations with batch size 2 on a single GPU for all the experiments. We adopt AdamW [21] as the optimizer with an annealing learning rate starting at 0.0001. We also adopt the exponential-moving-average (EMA) during evaluation and reconstruction. For the unified model, we train a single diffusion model on all the categories of MVTec-AD for 20,000 iterations. The class label is provided to the UNet [24] of the diffusion models for image reconstruction.

We resize the image to $(256, 256)$ and train the model with a 5 degree random rotation augmentation. For the pre-trained deep networks, we choose EfficientNet [30] pre-trained on ImageNet [9]. We set the ensembling factor α in 11 to 0.5 for all categories for the same-hyperparameter setting. Our best results are achieved with ensemble factors adjusted for each category. We select three timesteps $T = \{5, 50, 100\}$ during the forward diffusion process to get three different noise scales. The anomaly scores predicted are averaged as the final output. We set the learning rate for the gradient denoising process to be 0.02. The image is denoised once every $N_d = 5$ iteration.

4.2. Quantitative Results

We compare our anomaly localization results with CutPaste [19], Spade [6], PaDiM [8], DRAEM [40], CFlow [13], and UniAD [36]. We evaluate the results with two localization metrics: pixel-wise AUROC and Per-Region-Overlap (PRO) [3]. We present the results on the MVTec-AD benchmark in Tab. 1. Our model with the same hyperparameters for all categories boosts the PRO by 0.7%, and our best model with adjusted hyperparameters for each category improves the PRO by 1.1%, compared with previous state-of-the-art reconstruction-based methods. We show results on BTAD [22] in the supplementary.

Robustness. Our study demonstrates the effectiveness of incorporating random noise to enhance the robustness of anomaly localization. We observed that the previous state-of-the-art reconstruction method DRAEM [40] uses pseudo anomalies that are unsuitable for detecting anomalies that vary from the pseudo data, particularly in the cable, pill, and transistor classes. The UniAD's transformer-based AutoEncoder [36] performs poorly on the metal nut, tile, and wood classes. In contrast, our denoising model learns the normal data distribution for anomaly detection, which is robust for all the categories.

4.3. Qualitative results of localization.

Figure 6 shows the anomaly localization results on MVTec-AD [3]. The first and fifth columns are images with anomalies from MVTec-AD. The columns from left to right