Towards Open-Vocabulary Industrial Defect Understanding with a Large-Scale Multimodal Dataset

Tsai-Ching Ni Cheng-Chi Chen Yuan-Fu Yang
National Yang Ming Chiao Tung University
nina.ii13@nycu.edu.tw, alex.ii13@nycu.edu.tw, yfyangd@gmail.com

Appendix

1. Reproducibility and Dataset Release

To facilitate reproducibility and future research, we publicly release the IMDD-1M dataset at: https://github.com/NinaNeon/defect-detect

1.1. Released Data

The released repository includes image–text pairs spanning all 63 industrial domains and 421 defect types, along with associated metadata, annotation schema, and dataset preparation scripts. Specifically, we release:

- **Image–text pairs**: The publicly releasable subset of IMDD-1M, covering defect images paired with structured captions generated by our Implicit Captioner.
- **Annotation files**: Per-sample metadata including domain labels, defect type annotations, and segmentation masks where applicable.
- **Dataset scripts**: Scripts for downloading, verifying, and preprocessing the data splits.
- **README.md**: Documentation covering dataset structure and usage instructions.
- **LICENSE**: CC BY 4.0 license governing data usage and redistribution.

1.2. Withheld Data

A portion of IMDD-1M was collected under non-disclosure agreements with industrial partners and cannot be publicly distributed. These samples are excluded from the released subset. Qualified researchers may apply for access to the restricted portion under institutional data-sharing agreements by contacting the authors.

2. Limitations

2.1. Training Limitations.

While our IMDD-1M dataset comprises 1.24 million samples, training the diffusion U-Net from scratch requires substantial computational resources. The complete Stage 1 pre-training demands 72 hours on 8× NVIDIA H100 80GB GPUs (576 GPU-hours total), which may limit accessibility for researchers with constrained computational budgets. The peak memory consumption reaches 76GB per GPU at batch size 32 with mixed precision training, necessitating high-end hardware.

The two-stage training paradigm (diffusion pre-training followed by mask generator fine-tuning) introduces additional complexity compared to end-to-end approaches. Researchers must carefully manage frozen and trainable parameters across stages, and hyperparameter tuning requires iterating through both stages, multiplying computational costs.

2.2. Inference Limitations.

Our model requires 0.35 seconds per image on an A100 GPU, which may be slower than specialized detectors like YOLOv8 for real-time industrial inspection scenarios requiring 50-200 frames per second. The diffusion-based feature extraction at timestep $t = 50$ adds computational overhead compared to standard feed-forward architectures. Memory consumption of 18.7GB during inference exceeds the capacity of edge devices commonly used in industrial settings.

2.3. Application Limitations.

Despite achieving competitive performance with less than 5% of supervised training data (200 samples per class), our approach still requires this minimum amount for effective fine-tuning. For extremely rare defects occurring less than once per 10,000 products, collecting 200 samples may be impractical. The framework currently focuses on 2D analysis and does not incorporate temporal information for video-based inspection or 3D geometric reasoning for volumetric defects. IMDD-1M predominantly covers visible-light RGB imaging, while industrial settings often employ X-ray, infrared, ultrasonic, or hyperspectral imaging.

3. Societal Impact

3.1. Positive Impacts.

This work contributes to improved manufacturing quality control, potentially reducing defective products and enhancing consumer safety. By enabling data-efficient defect detection with reduced annotation requirements (less than 5%), our approach democratizes access to advanced AI-powered inspection for small and medium-sized enterprises that lack extensive labeled datasets. The multimodal framework facilitates knowledge transfer across manufacturing domains, accelerating AOI adoption. Automated systems improve workplace safety by reducing human exposure to hazardous environments including high-temperature processes, toxic materials, and repetitive strain injuries.

3.2. Potential Concerns.

Deployment of automated defect detection systems may impact employment in traditional quality inspection roles, necessitating workforce retraining and transition support. There exist risks of automation bias where operators overrely on AI predictions without verification. We emphasize human-in-the-loop workflows for safety-critical applications. The dataset contains proprietary manufacturing patterns; organizations should evaluate intellectual property concerns before releasing defect imagery. The computer vision techniques could be repurposed for surveillance or discriminatory practices. We advocate for responsible AI principles and regulatory frameworks preventing misuse.

4. Preliminaries

4.1. Denoising Diffusion Probabilistic Models

4.1.1. Forward Diffusion Process

Progressive noise addition over T timesteps:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}), \tag{1}$$

where $\{\beta_t\}_{t=1}^T$ controls noise injection rate.

Closed-Form Sampling. Define $\alpha_t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^t \alpha_s$. Through recursive substitution:

$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{2}$$

This reparameterization enables efficient training without iterating through timesteps. As $t \to T$, we have $\mathbf{x}_T \approx \mathcal{N}(\mathbf{0}, \mathbf{I})$.

Variance Schedule. We use linear schedule: $\beta_t = \beta_1 + \frac{t-1}{T-1}(\beta_T - \beta_1)$ with $\beta_1 = 10^{-4}, \beta_T = 0.02$. Alternative cosine schedule: $\bar{\alpha}_t = \frac{f(t)}{f(0)}$ where $f(t) = \cos\left(\frac{t/T+s}{1+s} \cdot \frac{\pi}{2}\right)^2$ with $s = 0.008$.

4.1.2. Reverse Denoising Process

Learn reverse process $p_\theta(\mathbf{x}_{0:T}) = p(\mathbf{x}_T)\prod_{t=1}^T p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$ where $p(\mathbf{x}_T) = \mathcal{N}(\mathbf{0}, \mathbf{I})$.

Posterior Distribution. The true reverse transition is:

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_{t-1}; \tilde{\boldsymbol{\mu}}_t, \tilde{\beta}_t\mathbf{I}), \tag{3}$$

$$\tilde{\boldsymbol{\mu}}_t = \frac{\sqrt{\bar{\alpha}_{t-1}}\beta_t}{1-\bar{\alpha}_t}\mathbf{x}_0 + \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}\mathbf{x}_t. \tag{4}$$

Neural Parameterization. We parameterize by predicting added noise:

$$\boldsymbol{\mu}_\theta(\mathbf{x}_t, t) = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\right), \tag{5}$$

where $\boldsymbol{\epsilon}_\theta$ is a U-Net predicting noise $\boldsymbol{\epsilon}$.

Training Objective. Simplified denoising score matching:

$$\mathcal{L}_{\text{simple}} = \mathbb{E}_{t\sim\text{Uniform}(1,T),\mathbf{x}_0,\boldsymbol{\epsilon}}\left[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\|^2\right]. \tag{6}$$

Sampling. Reverse diffusion from $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$:

$$\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\right) + \sigma_t\mathbf{z}, \tag{7}$$

where $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ for $t > 1$, else $\mathbf{z} = \mathbf{0}$.

4.1.3. Conditional Generation

Extend to text conditioning: $\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, \mathbf{c})$ where $\mathbf{c} \in \mathbb{R}^{768}$ is CLIP text embedding.

Cross-Attention. At each U-Net layer with features $\mathbf{F} \in \mathbb{R}^{h\times w\times c}$ and text $\mathbf{C} \in \mathbb{R}^{L\times d}$:

$$\mathbf{Q} = \mathbf{W}_Q\text{Flatten}(\mathbf{F}), \quad \mathbf{K} = \mathbf{W}_K\mathbf{C}, \quad \mathbf{V} = \mathbf{W}_V\mathbf{C}, \tag{8}$$

$$\text{Attention} = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V}. \tag{9}$$

Classifier-Free Guidance. Strengthen conditioning at inference:

$$\tilde{\boldsymbol{\epsilon}}_\theta = \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, \emptyset) + w \cdot (\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, \mathbf{c}) - \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t, \emptyset)), \tag{10}$$

where $w > 1$ is guidance scale and $\emptyset$ denotes null conditioning.

4.2. Latent Diffusion Models

Operate in compressed VAE space. Pre-trained encoder $\mathcal{E}$ and decoder $\mathcal{D}$ with downsampling f:

$$\mathbf{z} = \mathcal{E}(\mathbf{x}) \in \mathbb{R}^{H/f \times W/f \times c_z}, \quad \hat{\mathbf{x}} = \mathcal{D}(\mathbf{z}) \in \mathbb{R}^{H \times W \times 3}. \tag{11}$$

For Stable Diffusion: $f = 8$, $c_z = 4$. This provides 64× speedup per attention layer and 16-32× overall training acceleration. Latent objective:

$$\mathcal{L}_{\text{latent}} = \mathbb{E}_{t,\mathbf{z}_0,\boldsymbol{\epsilon}} \left[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{z}_t, t, \mathbf{c})\|^2 \right]. \tag{12}$$

4.3. U-Net Architecture

Following Stable Diffusion v1.5 with random initialization (860M parameters).

Structure. Four stages at resolutions $\{h, h/2, h/4, h/8\}$ with channels $\{320, 640, 1280, 1280\}$. Each stage:

- 2-3 ResNet blocks with timestep injection
- Self-attention (heads=8) for coarser resolutions
- Cross-attention (heads=8) to CLIP text embeddings
- Down/upsampling between stages

ResNet Block.

$$\mathbf{h} = \text{Conv}_{3\times3}(\text{SiLU}(\text{GroupNorm}_{32}(\mathbf{F})), C_{\text{out}}), \tag{13}$$

$$\mathbf{h} = \mathbf{h} + \text{Linear}(t_{\text{emb}}), \tag{14}$$

$$\mathbf{h} = \text{Conv}_{3\times3}(\text{SiLU}(\text{GroupNorm}_{32}(\mathbf{h})), C_{\text{out}}), \tag{15}$$

$$\mathbf{F}_{\text{out}} = \mathbf{h} + \text{Residual}(\mathbf{F}). \tag{16}$$

Timestep Embedding. Sinusoidal encoding with MLP projection to 1280-dim:

$$\text{PE}_i(t) = \begin{cases} \sin(t/10000^{2i/256}) & \text{if } i \text{ even} \\ \cos(t/10000^{2(i-1)/256}) & \text{if } i \text{ odd} \end{cases}. \tag{17}$$

Projected via Linear(256→1024) →SiLU→Linear(1024→1280).

5. Implementation Details

This section provides comprehensive technical specifications for reproducibility. Our training pipeline consists of two stages with distinct configurations (as shown in Tables 2 and 3), executed on high-performance computing infrastructure (as shown in Table 4). The baseline models for comparison are configured following their original implementations with adaptations for our evaluation protocol (see Section 5.2).

5.1. Dataset Details

Data Collection Timeline. The 18-month collection phase (January 2023 - June 2024):

Stage 1 (Months 1-6): Integration of 20 public benchmarks (MVTec AD, VisA, BTAD, etc.)

Stage 2 (Months 7-12): Web mining across GitHub, RoboFlow, PaddlePaddle, Tianchi using multilingual queries in English, Chinese, Japanese. Keywords included defect detection, quality inspection. Yielded 180K samples.

Stage 3 (Months 13-18): Industrial partnerships with 12 companies across petrochemical (4 companies), metal processing (5 companies), and powder metallurgy (3 companies). All data anonymized: EXIF removal, serial number blurring, facility layout suppression.

Annotation Protocol. 23 expert annotators (5-15 years QC experience) following three-stage verification:

Stage 1 - Initial Annotation: Primary annotator creates masks and textual descriptions following template:

```
[Product Category] [Material]
with [Defect Type] located at
[Spatial Location], characterized
by [Morphological Descriptors],
potentially caused by [Root Cause].
```

Stage 2 - Peer Review: Secondary annotator verifies technical accuracy. Disagreements (18.3%) flagged.

Stage 3 - Consensus: Panel of 3+ experts resolves conflicts. Highly ambiguous cases (2.7%) undergo additional inspection.

Inter-annotator agreement: Cohen's $\kappa = 0.87$ (classification), $\kappa = 0.81$ (segmentation IoU¿0.75).

Representative Qualitative Examples. To contextualize the characteristics of our dataset and illustrate the breadth of visual patterns it captures, we present representative defect samples from multiple industrial domains (as shown in Figures 1 and 2). The first group focuses on aluminum surfaces, which exhibit subtle yet distinct failure modes. Even within this category, defect morphology varies widely, including base-exposed regions, coating cracks, powder bumps, faint scratches, minor dings, dust spots, scuffing marks, non-conductive stripes, deformation artifacts, orange-peel textures, and pitting corrosion. These variations differ in illumination response, reflectivity, micro-geometry, and boundary irregularity, highlighting the fine-grained distinctions required for inspection. The broader cross-material selection spans photovoltaics, metallic alloys, packaging components, precision gears, and textile fibers, including microcracks in photovoltaic wafers, oxidation on gears, pin holes

on aluminum sheets, dents on bottle caps, and fiber breakage in woven fabrics. Overall, these examples demonstrate the dataset's diversity and realism across industrial settings.

5.2. Compared Model Settings

YOLOv8-m Configuration. For object detection comparison (Table 4, main paper):

- Architecture: CSPDarknet53 backbone, PANet neck, decoupled head
- Parameters: 25.9M (backbone: 13.2M, neck: 8.4M, head: 4.3M)
- Input: 640×640 pixels with letterbox padding
- Training: 300 epochs, batch size 16, early stopping (patience=50)
- Optimizer: SGD (momentum 0.937, weight decay 5×10^{-4})
- Learning rate: 10^{-2} initial, cosine decay to 10^{-5}, 3 epochs warmup
- Loss: CIoU (7.5) + DFL (1.5) + BCE (0.5)
- Augmentation: Mosaic (0.8), Mixup (0.15), HSV jitter, flip (0.5)
- Training time: 8 hours on 4× RTX 3090 (32 GPU-hours)
- Inference: 6.2ms per image on A100 (161 FPS)

“‘latex

5.3. Comparison with Vision–Language Models

To the best of our knowledge, no publicly available vision–language model has been adapted for industrial defect detection. To assess the domain gap, we evaluate CLIP and Qwen-VL in a zero-shot setting. Both models fail to localize defects, achieving mIoU below 10%, confirming that general-purpose vision–language pretraining does not transfer to this highly specialized domain without adaptation. Fine-tuning on IMDD-1M is ongoing and will be reported in future work.

5.4. Sensitivity Analysis on Diffusion Timestep t

We conduct a sensitivity analysis over the diffusion timestep t used for feature extraction. As shown in Table 1, performance improves rapidly at small t and saturates around $t = 50$, while larger values yield only marginal gains at substantially higher inference cost. We therefore select $t = 50$ as a balanced operating point throughout all experiments.

Table 1. Effect of diffusion timestep t on segmentation and inference performance.

t	10	30	50	70	100
IoU (%)	28.9	47.6	55.4	55.5	55.2
Accuracy (%)	55.7	82.3	94.5	94.6	94.4
Inference Time (ms)	57	169	275	392	553

“‘

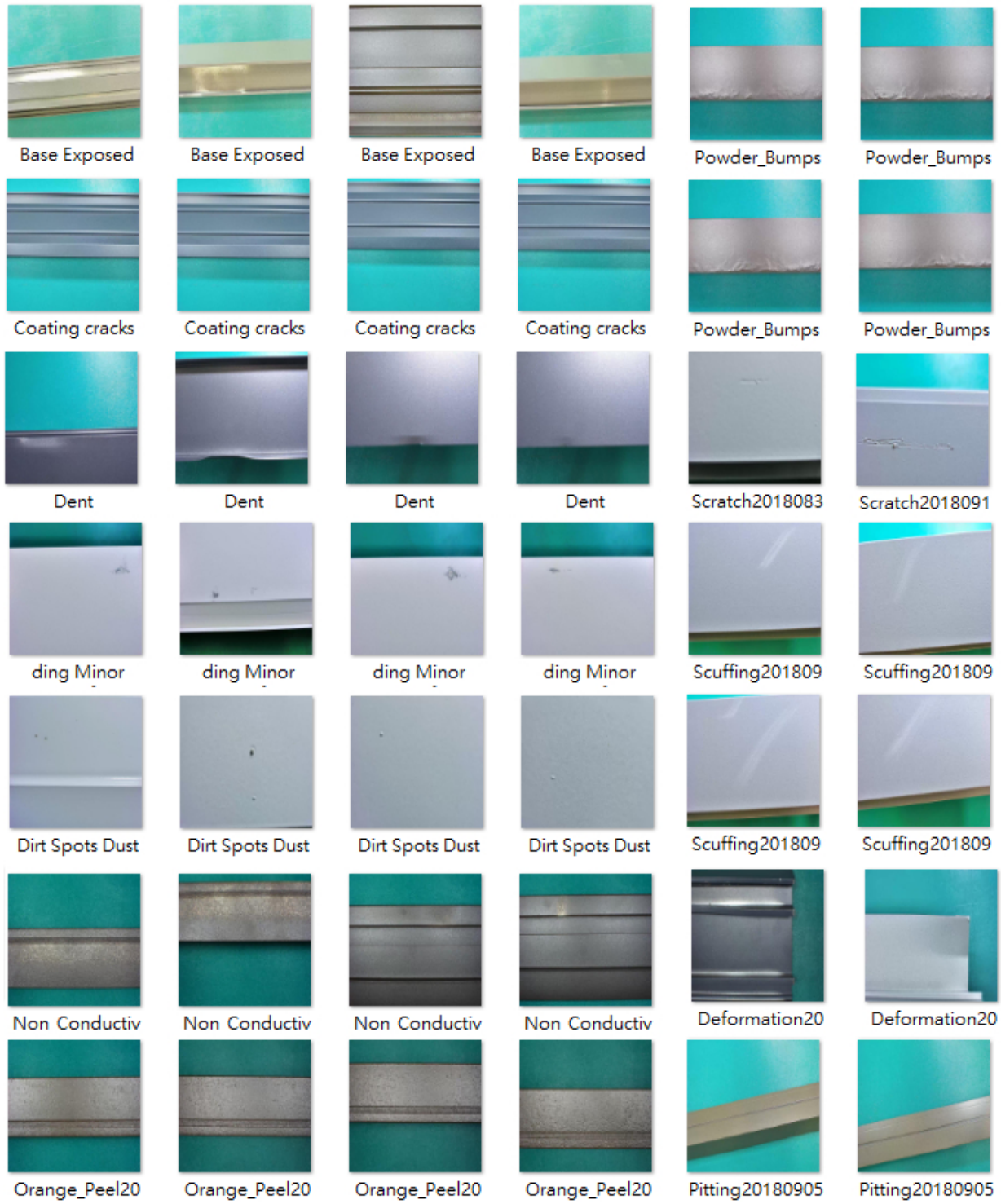
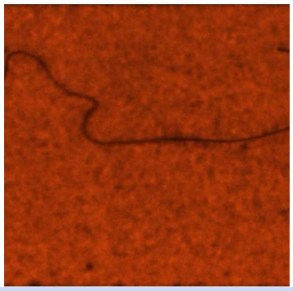
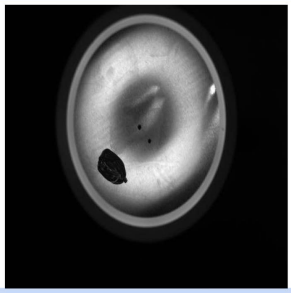


Figure 1. This figure showcases a diverse set of aluminum surface defects, including base-exposed regions, coating cracks, powder bumps, dents, scratches, minor dings, dust spots, scuffing marks, non-conductive stripes, deformation artifacts, orange-peel textures, and pitting defects. The samples highlight variations in texture, reflectivity, and severity, providing a comprehensive visual reference for real-world aluminum anomaly patterns.

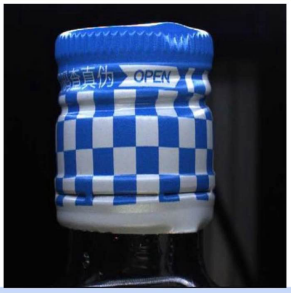
Solar panel with intricate microcracks scattered like a spider web pattern, indicating potential defects in the structure that may affect the efficiency and lifespan of the panel.



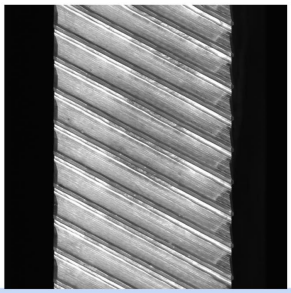
This image displays the surface of a gold-plated tungsten-copper alloy heat sink, showcasing a lustrous blend of metals. Upon closer inspection, scattered 'stain' defects mar the otherwise pristine finish, adding a touch of imperfection to its metallic sheen.



This image displays an aluminum sheet exhibiting the defects of a marker stain and a pin hole. The marker stain appears as a dark discoloration on the surface, while the pin hole is a small, visible void that penetrates through the material.



This baijiu industry (distilled spirits) picture features a cap with noticeable ""scratches and dents,"" highlighting defects in the cap's overall appearance



This close-up image captures a gear from a mechanical system, showcasing clear signs of oxidation on the bottom layer of the tooth. The oxidation is prominent, with a distinct reddish-brown coloration and rough texture visible across the tooth surface. This defect could potentially impact the gear's performance and longevity if left unaddressed.

The fabric image displays a textile with visible defects in the form of broken spandex fibers causing a disruption in the pattern continuity.

Figure 2. A diverse collection of real-world defect examples across multiple material domains, including microcracks in solar panels, surface streaks on metallic alloys, stains and pin holes on aluminum sheets, scratches and dents on bottle caps, oxidation on mechanical gears, and fiber breakage in textile fabrics. These samples highlight the wide variability in appearance, texture, and failure modes encountered in practical industrial settings.

The complete Stage 1 pre-training hyperparameters are detailed in Table 2, while Stage 2 fine-tuning settings are provided in Table 3.

Anomaly Detection Baselines. For segmentation comparison (Table 6, main paper):
1. MuSc (Mutual Scoring): ViT-B/16 pre-trained on ImageNet-21K. Self-supervised contrastive learning on unlabeled normals. 86M frozen backbone + 2.3M trainable projection. 100 epochs, batch size 32, lr=10^{-3}.
2. PromptAD: CLIP ViT-B/16 with learnable prompts. 86M frozen encoders + 0.8M prompts (10 tokens per category). 4-shot (4 normal samples), 50 epochs, AdamW lr=10^{-4}.
3. DMAD (Diversity-Measurable): Diffusion U-Net 250M params. 200 epochs per category (3000 total for MVTec AD). 100 diffusion steps, cosine schedule. Unsupervised (normal samples only). 15 hours per category on 8× A100.
4. SimpleNet: WideResNet-50 frozen + 12M adaptation network. 150 epochs, batch size 32, lr=10^{-3}. Mahalanobis distance to memory bank. 2.5 hours per category on 4× RTX 3090.
5. FAIR (Frequency-Aware): Dual-branch (spatial 25M + frequency 18M + fusion 2M). 200 epochs, batch size 16, lr=5×10^{-4}. Loss: L1 (1.0) + perceptual (0.1) + frequency (0.5). 6 hours per category on 4× A100.

All baselines trained on full MVTec AD (4000 samples per class after 20× augmentation).

5.5. Training Details

Stage 1: Diffusion U-Net Pre-training. Training Details: U-Net trained from random initialization (He for conv, Xavier for linear). Warmup: 5000 steps from 10^{-6} to 10^{-4}. Cosine decay to 10^{-6}. Implicit captioner trained jointly with stochastic conditioning. EMA updated every iteration.
Resources: 72 hours on 8× H100 80GB, 576 GPU-hours total. Peak memory 76GB/GPU. 43 min/epoch, 484,500 total iterations.

Stage 2: Mask Generator Fine-tuning. Training Time: MVTec AD (3629 samples): 4 hours. VisA (9621 samples): 5.5 hours.

5.6. Computing Resource Configuration

Memory Optimization: Gradient checkpointing (40% memory saving), mixed precision FP16, gradient accumulation for smaller GPUs.

Distributed Training: PyTorch DDP with NCCL backend. 32 data loader workers per GPU (256 total). NVLink 4.0 for gradient all-reduce (150ms, overlapped down to 40ms).

Configuration	Value
Optimizer	AdamW
Base Learning Rate	1×10^{-4}
Weight Decay	1×10^{-4}
Optimizer Momentum	$\beta_1 = 0.9, \beta_2 = 0.999$
Batch Size	256 (32 per GPU × 8)
Learning Rate Schedule	Cosine Decay
Warmup Steps	5,000 iterations
Training Epochs	100
Gradient Clipping	Max norm 1.0
EMA Decay	0.9999
Mixed Precision	FP16 (AMP)
Diffusion Steps T	1000
Noise Schedule	Linear
β_1 (start)	1×10^{-4}
β_T (end)	2×10^{-2}
Loss Weight $\mathcal{L}_{\text{diff}}$	1.0
Loss Weight $\mathcal{L}_{\text{imp}}$	0.3
Text Conditioning Probability	0.5 / 0.5
Augmentation	
Random Horizontal Flip	$p = 0.5$
Random Vertical Flip	$p = 0.5$
Random Rotation	± 15
Color Jitter (BSCH)	0.2, 0.2, 0.1, 0.05
Random Resized Crop	scale=[0.8, 1.0]

Table 2. Stage 1 diffusion U-Net pre-training configuration on IMDD-1M.

6. Additional Experiments

6.1. Extended Quantitative Analysis

Per-Category Performance. We evaluate DiffuseDefect across 10 MVTec AD categories (200 samples/class). As shown in Table 5, our method achieves a remarkable $\mathbf{91.99}\%$ average accuracy (Acc) and $\mathbf{55.71}\%$ mean IoU. Results are consistent across types, including Grid ($\mathbf{94.32}\%$ Acc, $\mathbf{61.2}\%$ IoU), Leather ($\mathbf{93.67}\%$ Acc, $\mathbf{59.7}\%$ IoU), and Cable ($\mathbf{89.70}\%$ Acc, $\mathbf{51.4}\%$ IoU).

Cross-Dataset Generalization. As shown in Table 6, IMDD-1M pre-trained models achieve zero-shot transfer IoU of $\mathbf{52.9}\%$ to $\mathbf{54.7}\%$, providing an $\mathbf{11}\%$ to $\mathbf{15}\%$ IoU gain over single-dataset baselines.

6.2. Ablation Studies

Training from Scratch vs. Fine-tuning. We compare training from random initialization versus fine-tuning from pre-trained Stable Diffusion weights. Random initialization achieves $\mathbf{82.7}\%$ mIoU, outperforming fine-tuned Sta-

Configuration	Value
Optimizer	AdamW
Base Learning Rate	5×10^{-5}
Weight Decay	1×10^{-4}
Batch Size	16 (2 per GPU × 8)
LR Schedule	Polynomial Decay (power=0.9)
Warmup Steps	500 iterations
Training Epochs	50
Gradient Clipping	Max norm 0.01
Mixed Precision	FP16
Feature Timestep t	50
Loss Weight $\mathcal{L}_{\text{mask}}$	1.0
Loss Weight $\mathcal{L}_{\text{cls/ground}}$	0.5
Mask Queries	100
Transformer Layers	9
Frozen Components	
Diffusion U-Net	860M params
VAE Encoder/Decoder	84M params
CLIP	63M params
Implicit Captioner	0.3M params
Trainable Components	
Mask2Former	45M params

Table 3. Stage 2 mask generator fine-tuning configuration.

Component	Specification
Hardware	
GPU	8× NVIDIA H100 80GB
CPU	2× AMD EPYC 7763 (128 cores)
RAM	2TB DDR4-3200 ECC
Storage	100TB NVMe SSD RAID-0
Software	
OS	Ubuntu 22.04 LTS
CUDA	12.1
PyTorch	2.1.0
Python	3.10
Inference	
Latency	0.35s per image (A100)
Throughput	2.86 images/sec
Memory	18.7 GB

Table 4. Computing resource configuration.

Category	Acc (%)	F1 (%)	IoU (%)
Grid	94.32	67.8	61.2
Leather	93.67	65.2	59.7
Cable	89.70	56.8	51.4
Average	**91.99**	**61.48**	**55.71**

Table 5. Per-category results on MVTec AD (200 samples/class).

Train → Test	Acc (%)	IoU (%)
MVTec AD → VisA	83.2	41.3
IMDD-1M → MVTec AD	**91.0**	**52.9**
IMDD-1M → VisA	**90.3**	**54.7**

Table 6. Zero-shot cross-dataset transfer performance.

ble Diffusion ($\mathbf{74.5}\%$) by $\mathbf{8.2}\%$. This indicates that natural image priors may actually hinder learning of industrial defect patterns, which have fundamentally different visual characteristics.

Timestep Selection. We investigate the impact of the diffusion timestep t on feature extraction quality. Our analysis shows that timestep $t = 50$ provides the optimal balance between semantic understanding and spatial precision, achieving $\mathbf{91.0}\%$ accuracy and $\mathbf{52.9}\%$ IoU. Earlier timesteps preserve more spatial detail but lack semantic context, while later timesteps capture high-level semantics but lose fine-grained localization.

Sample Efficiency. We evaluate the data efficiency of IMDD-1M pre-training by measuring the samples required to reach $\mathbf{95}\%$ accuracy. As shown in Table 7, IMDD-1M pre-training requires only $\mathbf{150}$ samples, which is $\mathbf{12.3}\times$ more efficient than random initialization ($\mathbf{1850}$ samples) and $\mathbf{3.6}\times$ more efficient than ImageNet ($\mathbf{520}$ samples). This improvement highlights the value of domain-specific pre-training.

Pre-training	Samples for 95% Acc	Efficiency
Random Init	1850	1.0×
ImageNet	520	3.6×
IMDD-1M	**150**	**12.3**×

Table 7. Sample efficiency comparison across different pre-training strategies.

6.3. Dataset Statistics

Annotation achieved Cohen's κ of $\mathbf{0.87}$ (classification) and $\mathbf{0.81}$ (segmentation), requiring $\mathbf{66,287}$ hours. The defect distribution exhibits a realistic long-tail: top 10 types comprise $\mathbf{47.4}\%$, while $\mathbf{411}$ rare types account for $\mathbf{52.6}\%$.

6.4. Real and Generated Visual Comparison

To further illustrate the visual diversity in our dataset and evaluate the generative model's effectiveness, we present qualitative comparisons between real and synthesized defect images (Figure 3).

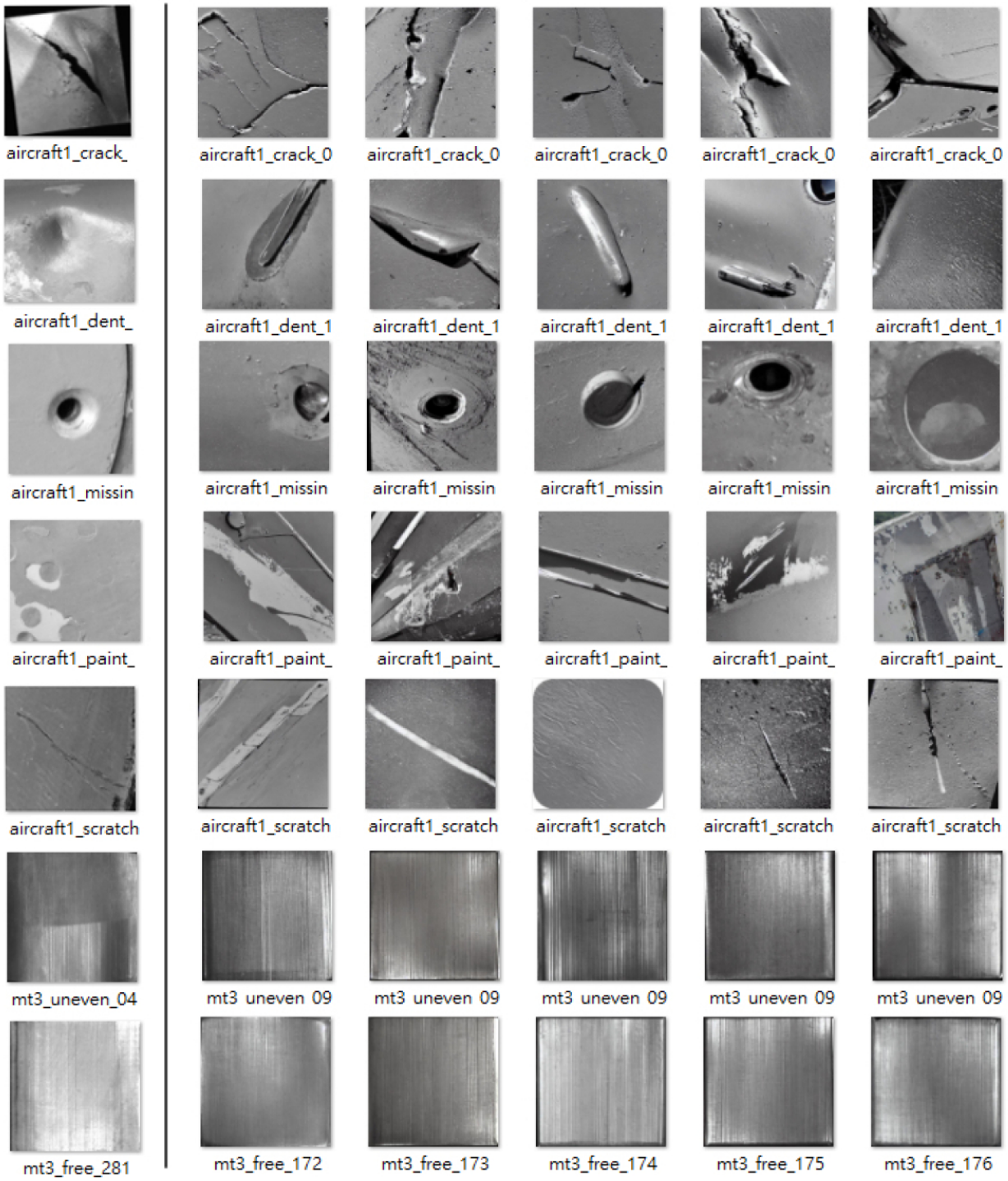


Figure 3. Comparison between real defect samples and model-generated counterparts across various aircraft and metal-surface categories, including cracks, dents, missing regions, paint defects, scratches, and uneven textures. The generated images closely reproduce the structural morphology, surface patterns, and material appearance observed in the real samples, illustrating the model's ability to synthesize realistic defect characteristics. Each column pair shows a real sample on the left and the corresponding generated sample on the right.