Concrete Crack Anomaly Detection with Diffusion Model

KENTECH Graduate School Winter Research Internship

January 2026

Hyunseok Lee[1] **Gangmin Lee**[2]

[1]Department of Data Science, Sejong University, Korea

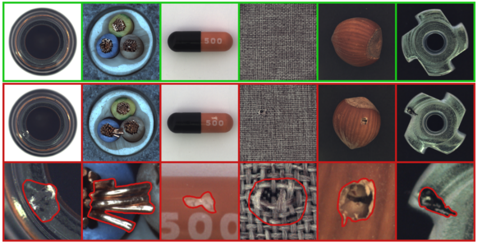
[2]Department of Digital media, Ajou University, Korea

Contents

Contents

Nature of Anomalies

- **Definition:** Rare, unexpected events differing from the majority (e.g., industrial defects, medical tumors).
- **Key Challenge:**
  - Impossible to define all anomalies.
  - Anomalies are *heterogeneous* and *unknown*.
- **Critical Risks:** Missing a single anomaly leads to catastrophic consequences (Cost, Safety).



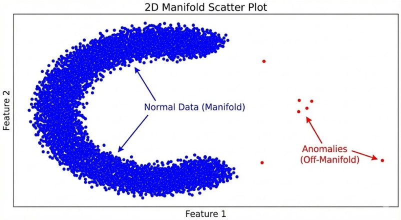
(a) Bottle (b) Cable (c) Capsule (d) Carpet (e) Hazelnut (f) Metal nut

Approaches to Anomaly Detection?

1. **Reconstruction-based**
   - Assumption: Anomalies cannot be reconstructed well.
   - Models: AE, VAE, Diffusion Models

2. **Representation-based**
   - Embedding features into a latent space to measure distance.
   - Models: One-class SVM, PatchCore.

3. **Probabilistic-based**
   - Estimating the density of normal data.
   - Models: Normalizing Flows.



Visualizing the manifold hypothesis

Why Unsupervised?

Data Imbalance

- Anomalies are naturally rare in optimized systems.
- Obtaining labeled anomalous data is expensive or impossible.

Unknown

- Supervised models overfit to known defects but fail on novel, unseen patterns (Open-set problem).
- Anomalies are infinite in variety; we cannot define them all.

$\Longrightarrow$ **Therefore, we must learn "What is Normal" (Unsupervised).**

Contents

Reconstruction-based Anomaly Detection

Core Hypothesis

- Deep neural networks trained *only* on normal data learn the **manifold of normal patterns**.
- Anomalies do not belong to this manifold $\rightarrow$ fail to reconstruct.
- **Assumption**:

$$\mathcal{L}_{\text{recon}}(\text{Normal}) \ll \mathcal{L}_{\text{recon}}(\text{Anomaly})$$

Why Reconstruction-based?

- **One-Class Classification**:
  - Anomalies are infinite/unknown (Open-set).
- **Interpretability**:
  - Residual map ($|x - \hat{x}|$) provides localization.
- **Generalizability**:
  - Works on images/signals without labels.

Auto Encoder (AE): Baseline Model

- **Concept**: Unsupervised learning to reconstruct input via a bottleneck.
- **Objective**: Minimize reconstruction error $\mathcal{L} = ||x - \hat{x}||^2$.

Critical Limitation: Identity Mapping

- AEs often generalize *too well*.
- They can reconstruct anomalies perfectly, leading to **low anomaly scores** for actual defects.

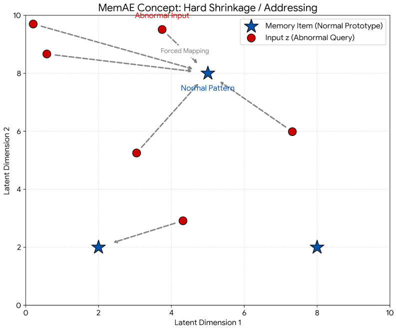
MemAE: Constraining the Latent Space

Memory-augmented AutoEncoder

- **Motivation**: Solve the “Identity Mapping” problem of AE.
- **Key Idea**: Memory Module

Mechanism

1. Encoder produces a query z.
2. Retrieve most relevant **Normal Prototypes** from Memory.
3. Decoder reconstructs using only normal patterns.



Anomalies are reconstructed as **Normal** $\rightarrow$ **High Error**.

DRAEM: Learning to Denoise

Discriminatively trained Reconstruction Anomaly Embedding Model

- **Motivation**: L2 loss is not sensitive enough to structural anomalies
- **Key Idea**: Synthetic Anomalies & Discriminative Learning.

Step 1: Simulation

- Generate synthetic defects (Perlin Noise) on normal images.
- "Just-out-of-distribution".

Step 2: Dual Tasks

- **Reconstructor**: Denoise the image.
- **Discriminator**: Segment the anomaly.

Moving beyond simple pixel-wise differences.

Limit of Deterministic Approaches

AE, MemAE, and DRAEM are **Deterministic**.

They map $x \rightarrow \hat{x}$, but do not explicitly model the **Probability Distribution** $p(x)$.

$\Downarrow$

Shift to Probabilistic Generative Models

Variational Auto Encoder(VAE)

- **Concept**: Learns the Likelihood of data, not just reconstruction.
- **Evidence Lower Bound (ELBO)**:

$$\log p(x) \geq \underbrace{\mathbb{E}_{q(z|x)}[\log p(x|z)]}_{\text{Reconstruction Term}} - \underbrace{D_{KL}(q(z|x)||p(z))}_{\text{Regularization Term}}$$

Why VAE?

- Introduces **Reconstruction Probability** instead of Error.
- Provides the **Mathematical Foundation** for Diffusion Models.

Contents

Diffusion Model

- **Analogy**: "Destroying data structure slowly, then learning to reverse time."
- **Key Insight:** Generating data $\approx$ Iteratively removing noise.

1. Forward Process (Diffusion)

- Gradually add Gaussian noise to data x_0 until it becomes pure noise $x_T \sim \mathcal{N}(0, I)$.
- *Fixed*, parameter-free Markov chain.

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I)$$

2. Reverse Process (Denoising)

- Learn to remove noise to restore structure.
- Modeled by a neural network $\epsilon_\theta(x_t, t)$.

$$p_\theta(x_{t-1}|x_t) \approx \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta)$$

1. DDPM

- **Goal**: Train the network ϵ_θ to predict the noise added to the image.
- **Simplification**: The complex ELBO objective simplifies to a weighted MSE Loss.

Training Objective ($\mathcal{L}_{\text{simple}}$)

$$\mathcal{L}_{\text{simple}}(\theta) := \mathbb{E}_{t,x_0,\epsilon}\left[\|\epsilon - \epsilon_\theta(\underbrace{\sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon}_{\text{Noisy Input } x_t}, t)\|^2\right]$$

- **Limitation of DDPM**:
  - **Slow**: Requires simulating the Markov chain for $T \approx 1000$ steps.
  - **Stochastic**: Generates different outputs for the same input.

2. DDIM

DDIM Innovation

- **Non-Markovian Process**: Generalizes DDPM to skip steps (e.g., 1000 $\rightarrow$ 50 steps).
- **Deterministic Sampling**: Sets noise variance $\sigma_t = 0$.

Benefits for Anomaly Detection

- **Consistency**: Given the same latent x_T, it always reconstructs the exact same x_0.
- **Stability**: Reduces random noise artifacts in the residual map $|y - \hat{x}|$.

Result: Up to 50x Faster inference with Stable Reconstruction.

Learning Normality via Denoising (I): Mechanism

- **Training Objective**: Train the model ϵ_θ *only* on **Normal Concrete**.
- **Inference Hypothesis**:
  - Model learns the gradient $\nabla_x \log p(\text{Normal})$.
  - It projects *any* input onto the **Normal Manifold**.

The "Healing" Process

1. **Corrupt (Add Noise)**: Inject noise to input y to reach state x_t.

$$x_t = \sqrt{\bar{\alpha}_t} y + \sqrt{1 - \bar{\alpha}_t} \epsilon$$

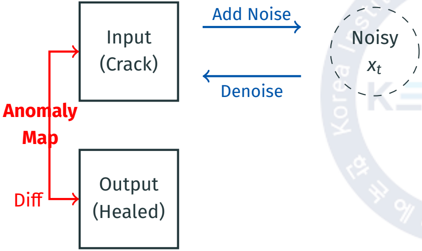
2. **Restore (Denoise)**: Denoise x_t using trained priors.

$$\hat{x}_0 \leftarrow \text{Reverse Process}(x_t)$$

3. **Result**: The anomaly disappears (Low probability in $p(\text{Normal})$).

Learning Normality via Denoising (II): Visualization

Visualizing the Restoration Process



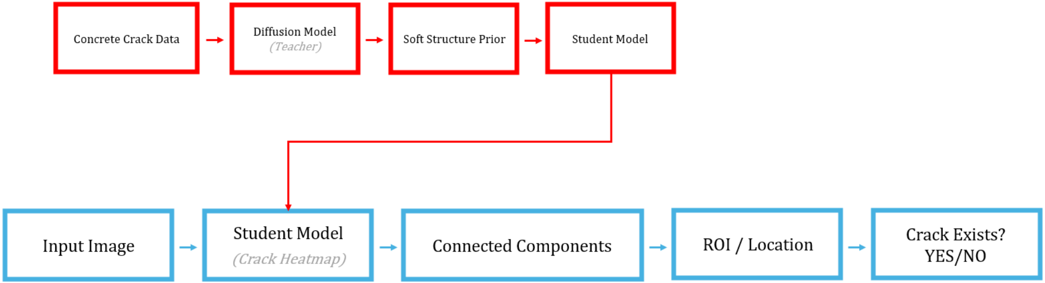
Why Diffusion is Superior?

- **High-Fidelity**:
  - Preserves **high-frequency texture** unlike blurry AEs.
- **Iterative Refinement**:
  - Step-by-step denoising allows precise correction.

Anomaly Score Design

Contents

Project Pipeline



- DataSet: https://www.kaggle.com/datasets/oluwaseunad/concrete-and-pavement-crack-images?select=Positive

Failure Cases

- **Extremely Thin or Faded Cracks**
  - Crack signals are too weak to be consistently preserved during denoising.
  - Both the diffusion teacher and the student model fail to capture sufficient structural evidence.
- **Stains, Rust, and Water Marks**
  - Non-structural linear patterns visually resemble cracks.
  - These patterns do not correspond to actual structural damage.
- **Severe Domain Shift**
  - Diffusion priors are learned from concrete crack datasets.
  - Heavily weathered or atypical utility poles deviate from this manifold.

Discussion Questions

- **Model Design Perspective**
  - Why is it effective to use diffusion models as a training-time teacher rather than a final predictor?
  - What advantages does soft supervision (e.g., heatmap-based targets) provide over binary labels in crack detection tasks?

- **Generalization**
  - Can this approach be extended beyond crack detection to other structural defects, such as corrosion, surface scratches, or similar line-like damages?

Timeline

Week 1~2	Week 3	Week 4
1. Implement a toy DDPM using crack skeletons or heatmap representations. 2. Visualize the forward and reverse denoising processes **Key question:** Does diffusion memorize pixels, or does it learn structural representations?	1. Use diffusion outputs (soft heatmaps) as teacher signals 2. Train a lightweight student model 3. Observe the first measurable improvement in crack presence classification performance	1. Compare DDPM and DDIM under the same supervision setup 2. Evaluate performance under different sampling step configurations **Key question:** Is stochasticity essential, or is iterative refinement the true core of diffusion?
Week 5	**Week 6**	**Week 7**
1. Use diffusion during training only 2. Remove diffusion entirely at inference time 3. Verify performance retention and significant efficiency gains	1. Introduce a coarse-to-fine student architecture without diffusion 2. Compare against single-step prediction models 3. Validate the architectural generalization of diffusion's core idea	1. Summarize the role of diffusion 2. Report final results and analyze remaining limitations 3. Discuss implications for practical inspection systems

Thank you for your attention!

Any Feedback is Welcome!