145th Master's Thesis Defense
1st Semester, 2026

A Study on Industrial Anomaly Localization based on Class-Conditioned Irredundant Feature Patch Selection

클래스 조건부 비중복 특징 패치 선별을 이용한 산업 이상 위치 탐지 연구

Se-Hoon Jung

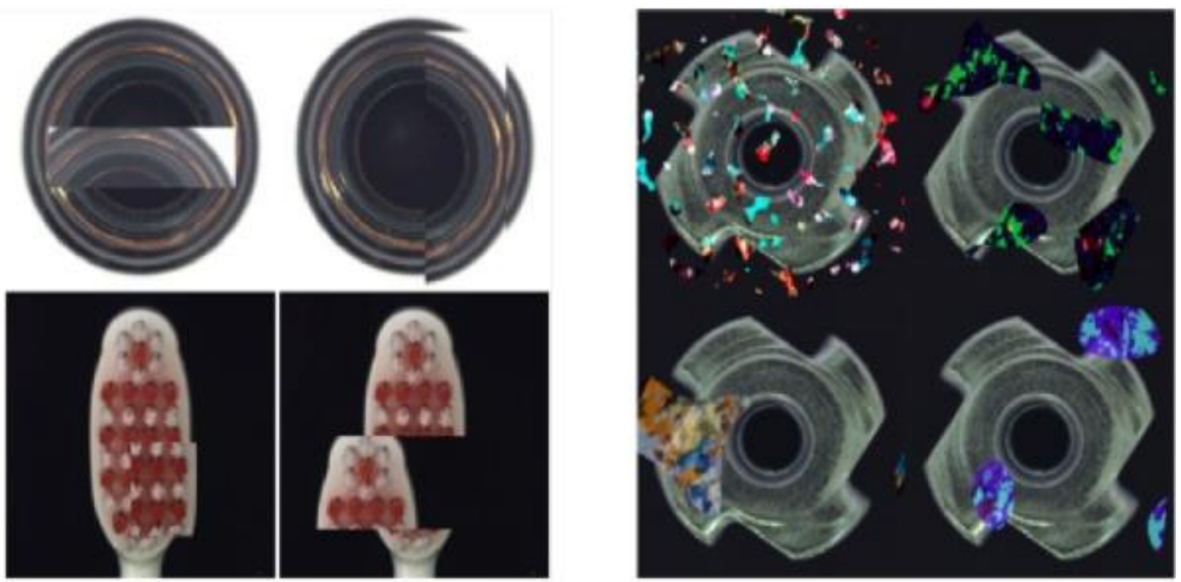
storyfor980107@gmail.com

Contents

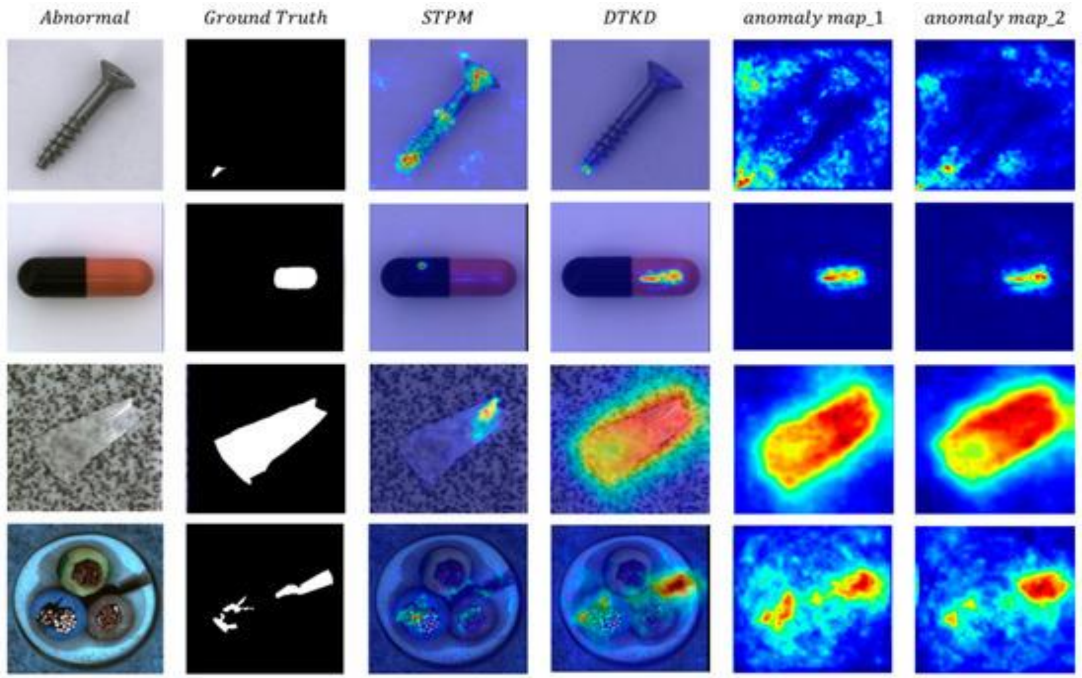
Introduction

What is Anomaly Localization?

- In manufacturing across electronics, automotive, battery, and pharmaceuticals, anomaly localization pinpoints defect locations in real time to cut rework, scrap, line stops, and warranty risks while enabling traceable quality gates and closed-loop control [5].
- Anomaly localization generates pixel-wise anomaly maps that highlight the precise spatial boundaries of defects, surpassing image-level detection in challenge under repetitive cluttered textures and low-contrast conditions [2].



[Figure 1] Manufacturing Real-Time Defect Localization Example



[Figure 2] Pixel-Level Anomaly Map Example

[2] A.-T. Ardelean et al., “Blind localization and clustering of anomalies in textures,” in Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR), 2024.
[5] Zhang, Zilong, et al. "Industrial anomaly detection with domain shift: A real-world dataset and masked multi-scale reconstruction." Computers in Industry 151 (2023): 103990.

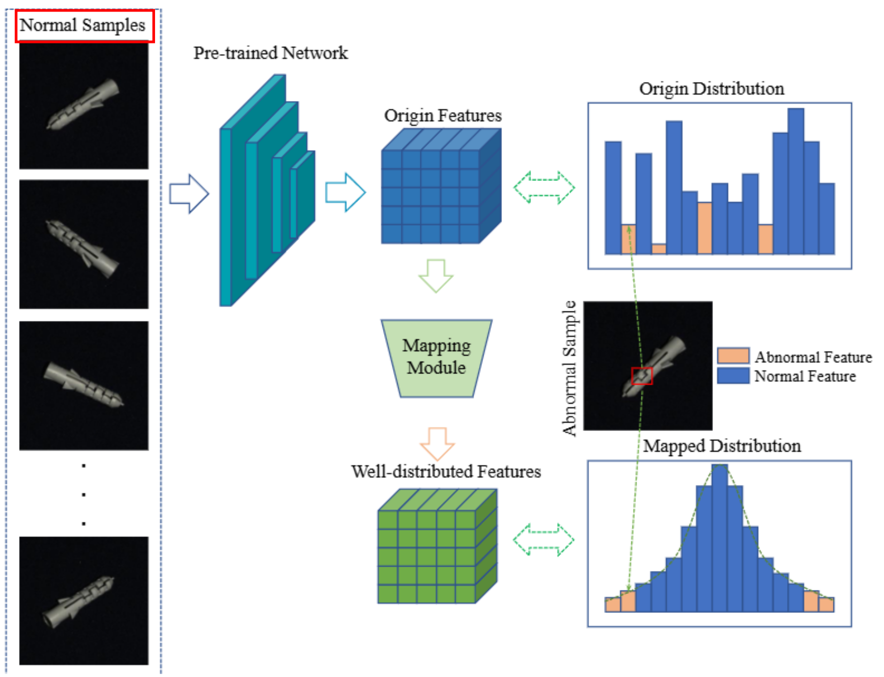
Introduction

Patch-based Anomaly Localization

- The patch-based anomaly localization (PAL) divides the feature map into spatial patches and calculates an anomaly score for each by measuring similarity to reference feature patches extracted from defect-free training samples.
- Considering its suitability for industrial practice, in this thesis, PAL is considered because of its promising performance for identifying defective regions, even though it does not require defective cases during the training phase.

Table 1. Taxonomy of Anomaly Localization Approaches

Strategy	Core Mechanism / Principle
Feature Retrieval-based	Compare test patch features with normal patch references stored in the memory bank.
Pixel Reconstruction-based	Train an autoencoder or GAN on normal data and detect by reconstruction error.
Feature Reconstruction-based	Reconstruct intermediate features and detect inconsistencies in reconstructed features.



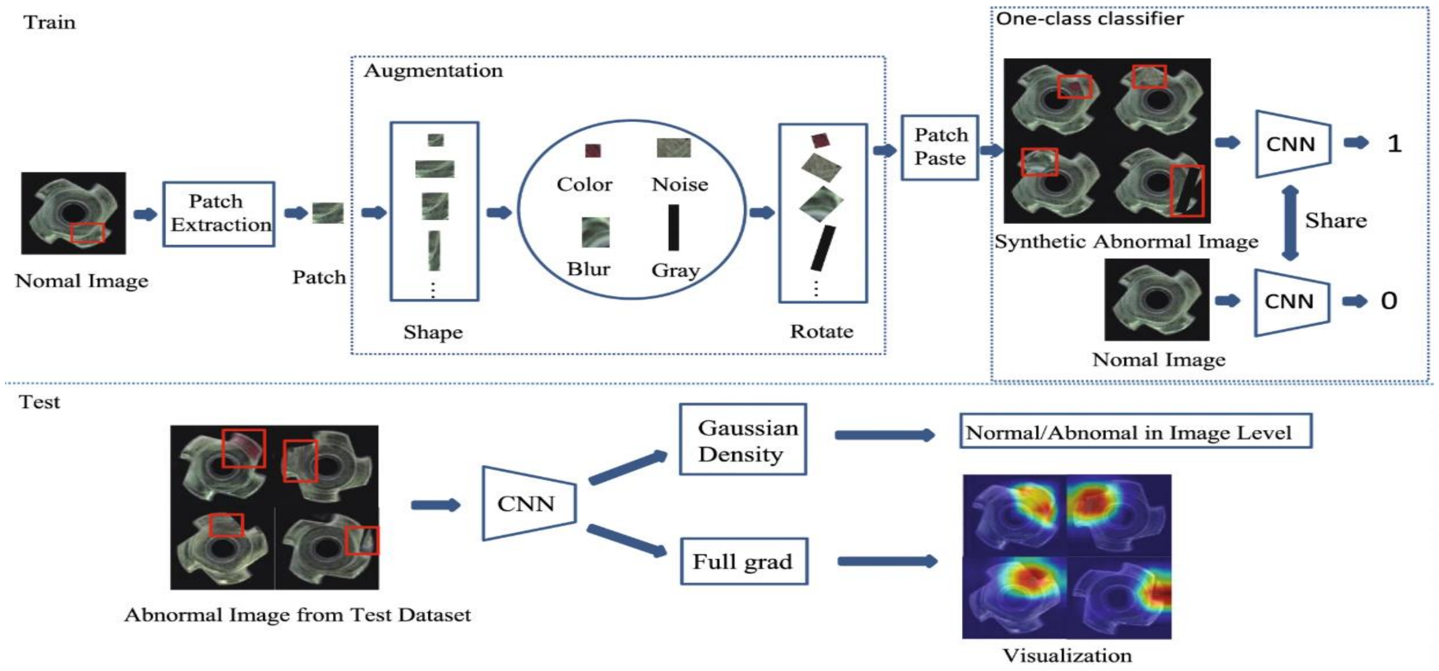
[Figure 3] Defect-free training using only normal samples.

[8] Simões, Francisco, et al. "Attention modules improve image-level anomaly detection for industrial inspection: A differnet case study." ." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), 2024

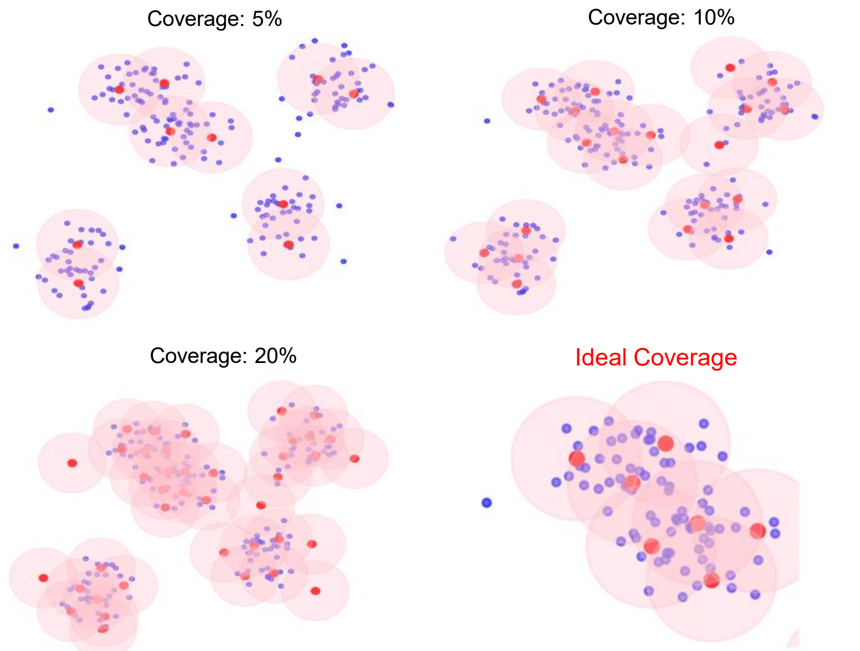
Introduction

Standard Procedure and Ideal Goal

- The standard procedure extracts multi-scale patch features, selects non-redundant representatives to build the memory bank, then at inference performs nearest-neighbor search with cosine distance and aggregates the distance values into a defect score.
- The ideal goal is to select diverse non-defective patch features from training data and store them in a memory bank that maximizes coverage, suppresses redundancy, and maintains reliable neighbor separation while keeping memory and latency controlled.



[Figure 4] PAL Anomaly Localization Process



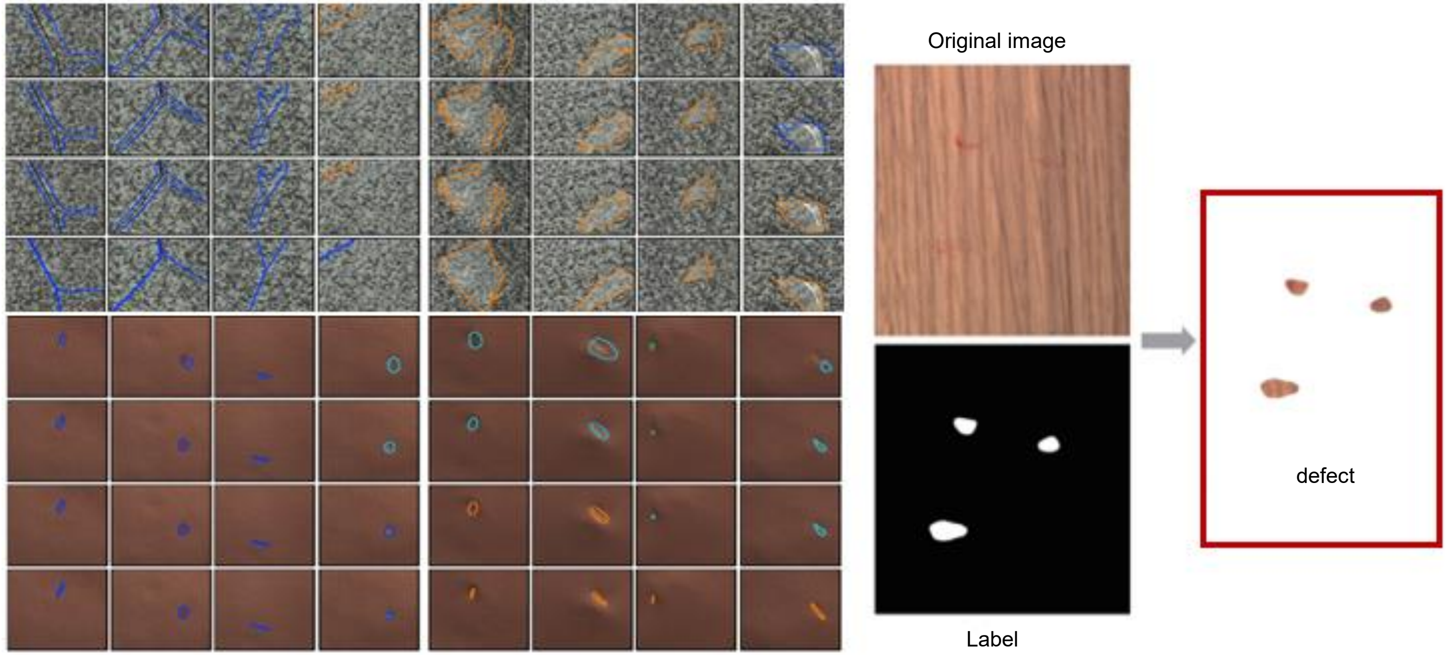
[Figure 5] Ideal Memory Bank Comparison

[2] A.-T. Ardelean et al., “Blind localization and clustering of anomalies in textures,” in Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR), 2024.
[6] M. Pei et al., “Self-supervised learning for industrial image anomaly detection by simulating anomalous samples,” International Journal of Computational Intelligence Systems, vol.16, no.1, p.152, 2023..
[7] T. Bao et al., “MIAD: A maintenance inspection dataset for unsupervised anomaly detection,” in Proc. of the IEEE/CVF Int. Conf. on Computer Vision (ICCV), 2023.

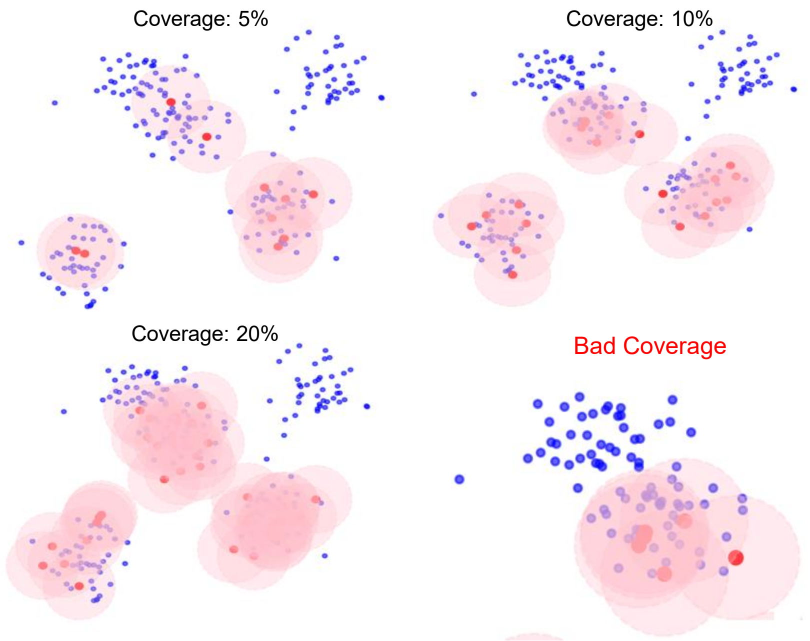
Introduction

Challenge

- In industrial settings, standardized processes produce repetitive patterns like woven fabrics, metal meshes, and circuit boards that resemble backgrounds, making it difficult for feature extractors to distinguish real defects from normal textures.
- When the training dataset contains repetitive patterns, learning acceptable variations can be disturbed because redundant feature patches overwhelm the memory bank, resulting in performance degradation due to excessively high false positives to normal pixels.



[Figure 6] Challenge images example

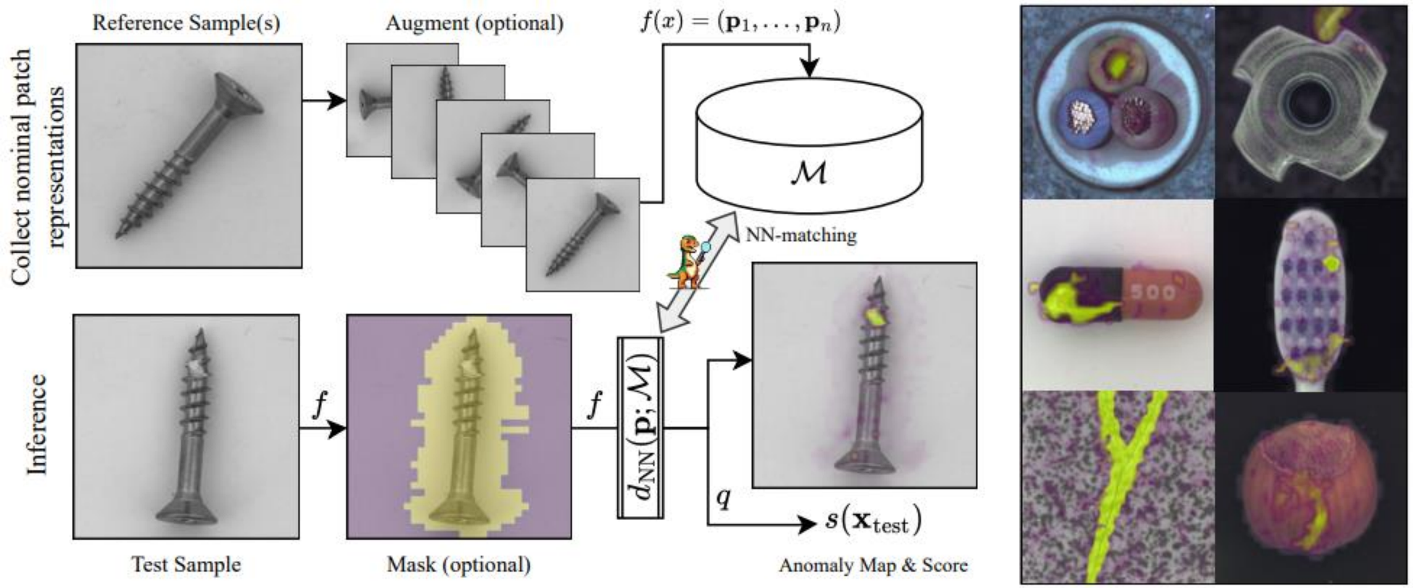


[Figure 7] Challenge images example 2

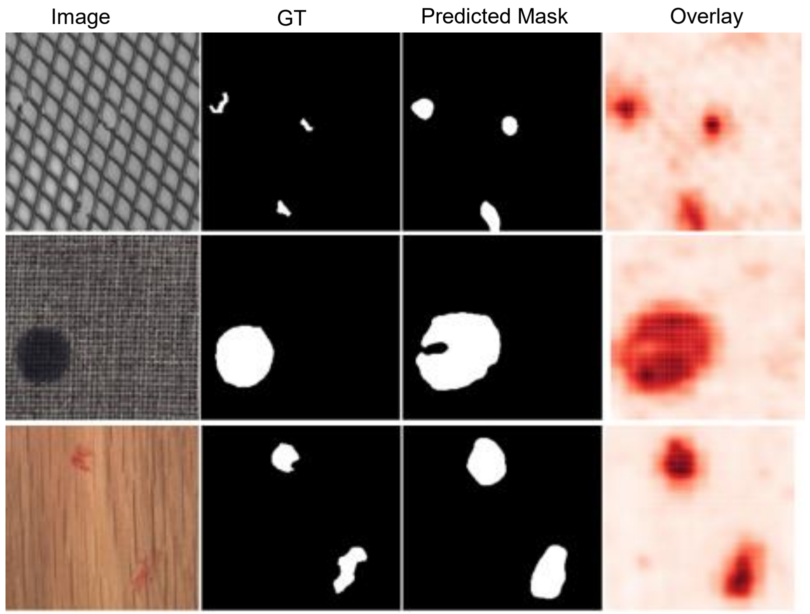
Related Work

AnomalyDINO: Boosting Patch-based Few-shot Anomaly Detection with DINOv2 [9]

- Representative feature patch selection extracts normal patches from DINOv2 ViT with spatial masking and rotation augmentation, builds the memory bank, measures patch similarity using cosine distance, and produces an anomaly score.
- Spatial masking and repeated rotation sample overlapping regions with similar textures lead to the selected patches to cluster into visually redundant representations and populate the memory bank with redundant foreground features.



[Figure 8] Overall framework of Anomaly DINO

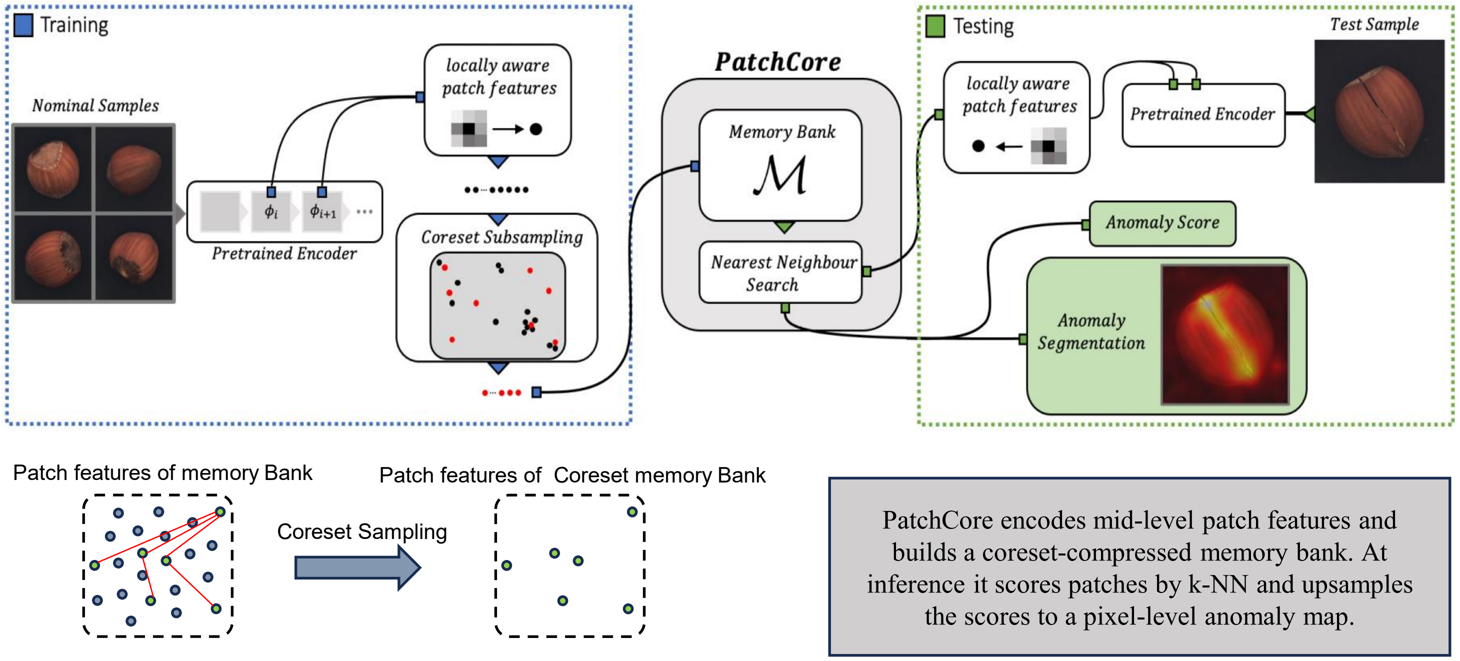


[Figure 9] Qualitative anomaly localization on MVTec AD

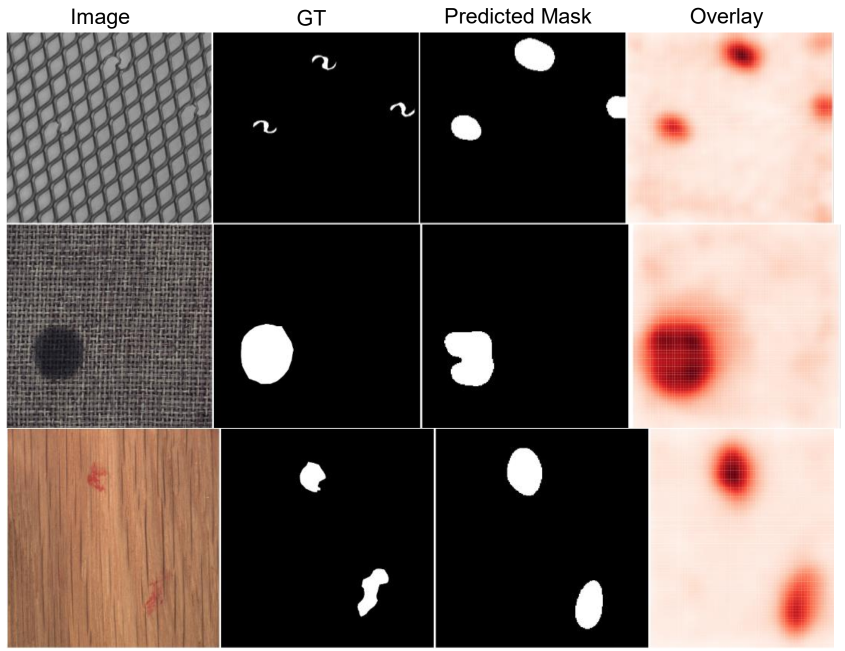
[9] Damm, Simon, et al. "Anomalydino: Boosting patch-based few-shot anomaly detection with dinov2." 2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). IEEE, 2025.

Towards Total Recall in Industrial Anomaly Detection [10]

- PatchCore constructs a compact memory bank by subsampling normal patch features to retain coverage while reducing redundancy, and computes anomaly scores by comparing each test patch with its nearest neighbors in the embedding space for precise localization.
- The feature patch selection strategy favors dense feature clusters, so rare or structurally unique patches are excluded, and the remaining representatives converge toward redundant patterns that limit the diversity of the memory bank.



[Figure 10] Overall framework of PatchCore



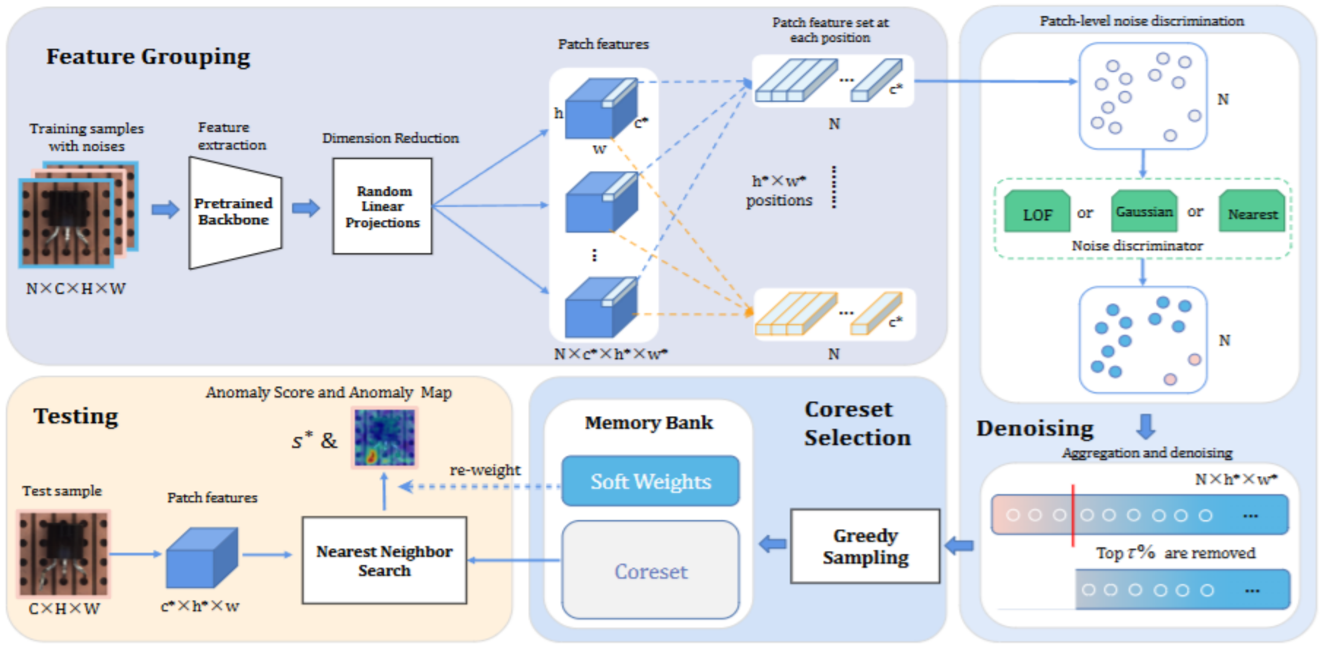
[Figure 11] Qualitative anomaly localization on MVTec AD

[10] Roth, Karsten, et al. “Towards Total Recall in Industrial Anomaly Detection.” Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022.

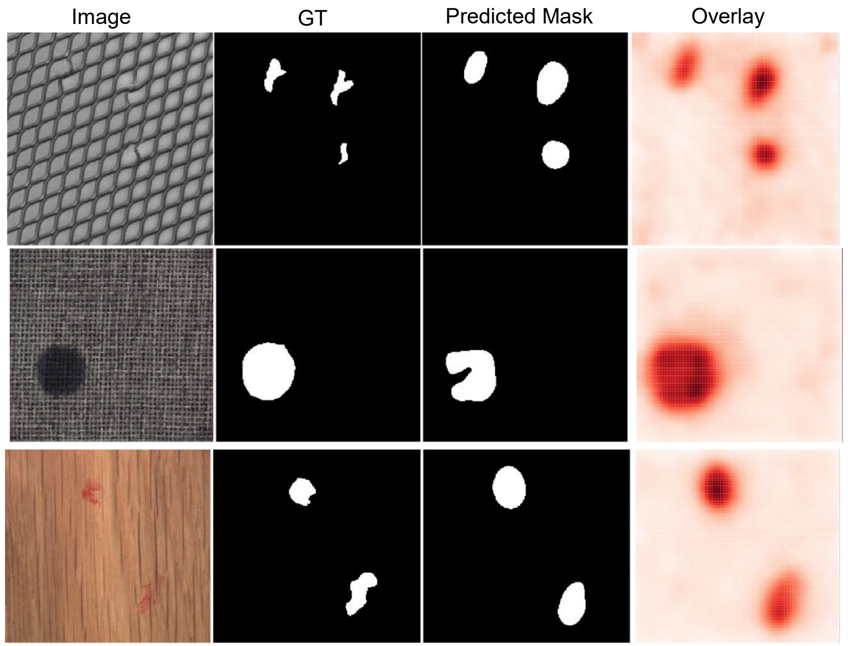
Related Work

SoftPatch: Unsupervised Anomaly Detection with Noisy Data [11]

- SoftPatch utilizes a noise discriminator to assign outlier scores and remove noisy patches, stores the refined patch features in the memory bank to soften the decision boundary, and performs weighted patch k-nearest neighbor for anomaly localization.
- Noise discrimination removes visually distinct but valid normal patches while emphasizing frequently clean textures, causing the retained features to represent limited variations and resulting in a memory bank with redundant feature patches.



[Figure 12] Overall framework of SoftPatch



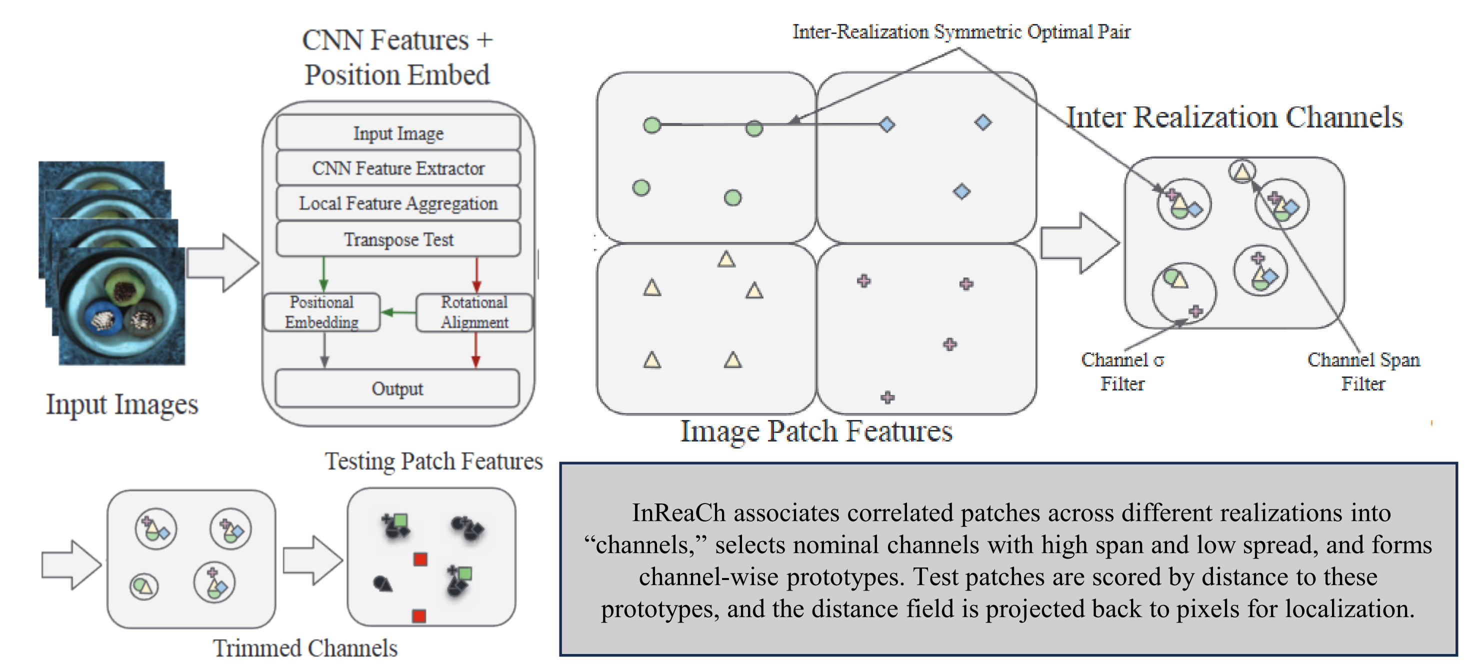
[Figure 13] Qualitative anomaly localization on MVTec AD

[11] Jiang, Xi, et al. "SoftPatch: Unsupervised Anomaly Detection with Noisy Data." Advances in Neural Information Processing Systems (NeurIPS), 2022.

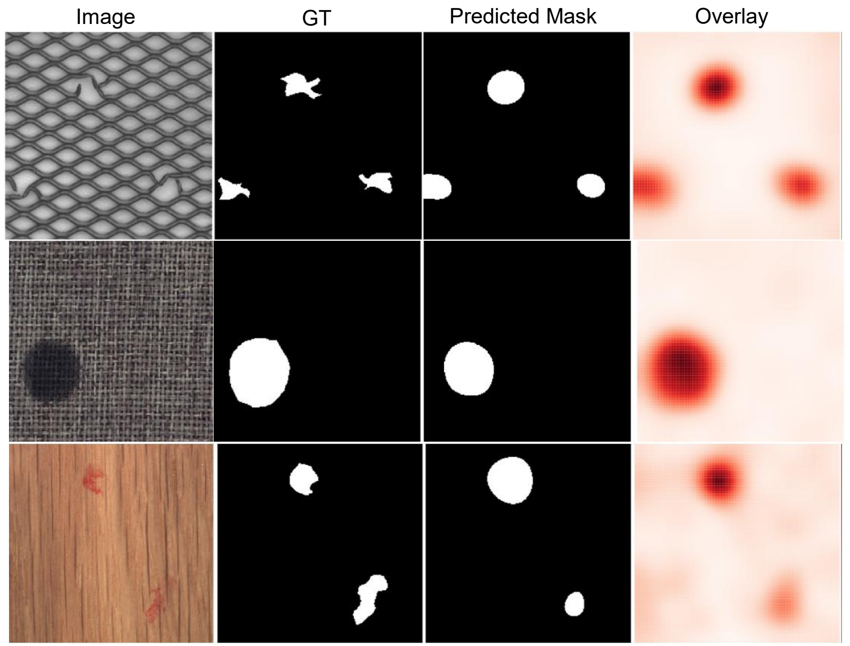
Related Work

Inter-Realization Channels: Unsupervised Anomaly Detection Beyond One-Class Classification [12]

- InReach selects channel-wise nominal prototypes by enforcing cross-image consistency between realizations, filters patch features through inter-realization channels, and computes patch-wise distances to the prototypes for fully unsupervised anomaly localization.
- The consistency constraint aligns correlated patches across realizations, producing prototypes that capture repetitive stable patterns and form a memory bank with redundant feature patches that underrepresents natural variability in normal data.



[Figure 14] Overall framework of Inter-Realization Channels



[Figure 15] Qualitative anomaly localization on MVTec AD

[12] McIntosh, Declan, and Alexandra Branzan Albu. "Inter-Realization Channels: Unsupervised Anomaly Detection Beyond One-Class Classification." Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2023.

Common Drawback and Solution

- Top p% feature patches contained in the memory bank can be redundant, causing performance degradation owing to an unintentional reduction in acceptable variation and limited representation of diverse normal textures.
- One direct solution for this issue is to select irredundant feature patches for each class based on similarity awareness, ensuring that the memory bank preserves diverse and distinctive normal patterns rather than simply including the top p% features.

Method	Venue	Summary	Common Drawback
Damm, Simon, et al [9]	WACV, 2025	A training-free, patch-based few-shot anomaly detector that builds a memory of normal DINOv2 patch embeddings and scores test patches via k-NN to yield smoothed anomaly maps and strong image-level detection.	Lack of class-aware diversity control leaves rare non-defective patterns uncovered under repetitive backgrounds, collapsing k-NN margins and triggering spurious activations with blurred boundaries.
Roth, Karsten, et al [10]	CVPR, 2022	Builds a coreset-selected memory of non-defective patch embeddings and localizes anomalies by nearest-neighbor distance.	
Jiang, Xi, et al [11]	NeurIPS, 2022	Extends PatchCore with patch-level denoising and soft weighting to prune or downweight noisy training patches and perform weighted k-NN detection.	
McIntosh, Declan, et al [12]	ICCV, 2023	Discovers cross-image patch channels via symmetric matching, keeps well-supported low-spread channels as nominal prototypes, and scores by distance to channel features.	

[9] Damm, Simon, Mike Laszkiewicz, Johannes Lederer, and Asja Fischer. "AnomalyDINO: Boosting Patch-Based Few-Shot Anomaly Detection with DINOv2." Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), 2025.
[10] Roth, Karsten, et al. "Towards Total Recall in Industrial Anomaly Detection." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022.
[11] Jiang, Xi, et al. "SoftPatch: Unsupervised Anomaly Detection with Noisy Data." Advances in Neural Information Processing Systems (NeurIPS), 2022.
[12] McIntosh, Declan, and Alexandra Branzan Albu. "Inter-Realization Channels: Unsupervised Anomaly Detection Beyond One-Class Classification." Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2023.

Proposed Method

Notation and Definition

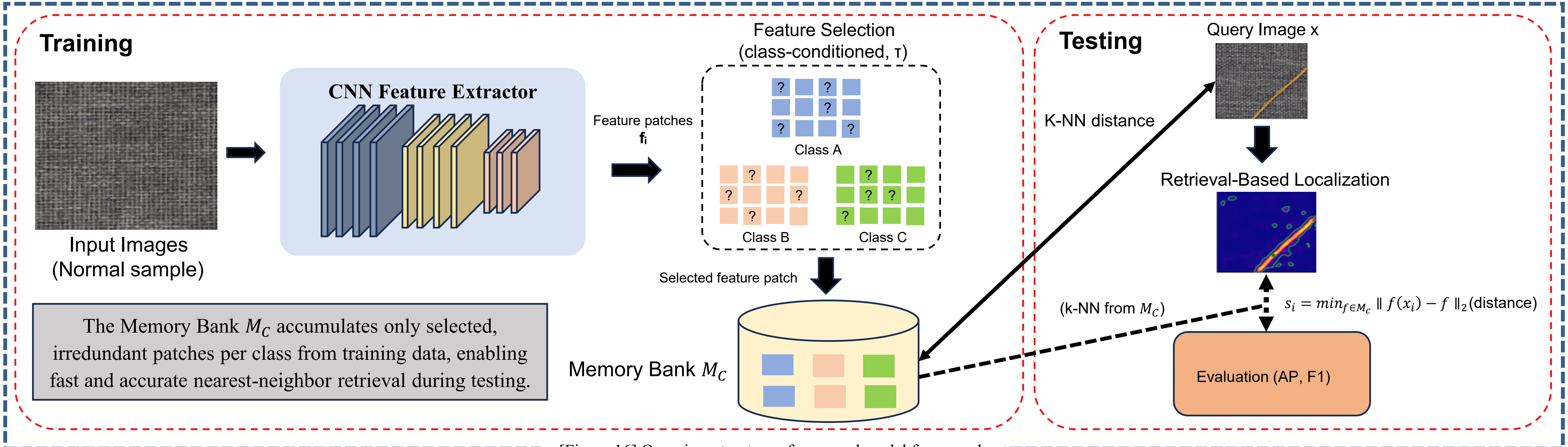
- The MVTec AD dataset is defined as $\mathcal{D} = \{(x_i, y_i, c_i)\}_1^N$, where x_i denotes an input image, y_i represents its pixel-wise ground truth mask, and c_i indicates the corresponding class label used for class-conditioned feature patch selection.
- During testing, the model constructs a class-specific memory bank M_C from non-defective training features and generates a predicted anomaly map $p(x_i)$ by computing patch-wise distances to M_C for localization.

Notation	Name	Definition
$\mathcal{D}$	Dataset	Entire MVTec AD dataset consisting of $\{(x_i, y_i)\}$ pairs
x_i	Input Image	Normalized input image, which may contain normal or anomalous regions
y_i	Ground Truth Mask	Pixel-wise binary mask indicating the location of anomalies
f_i	Feature Patch	Local feature embedding extracted from encoder backbone
N	Dataset Size	Total number of samples in the dataset
$p(x_i)$	Predicted Anomaly Map	Pixel-wise map computed by patch-to-memory comparison
M_C	Class-specific Memory Bank	Set of irredundant feature patches selected for class c
τ	Similarity Threshold	Criterion for filtering redundant patches during selection
AP	Average Precision	Metric measuring precision-recall area for pixel-wise anomaly detection
F_1	F1 Score	Harmonic mean of precision and recall used as auxiliary evaluation

Proposed Method

Graphical abstract

- In this presentation, a novel feature patch selection for the patch-based localization method is introduced to construct a memory bank composed of irredundant features, enabling more reliable anomaly localization across diverse industrial textures.
- Based on the memory bank M that is filled with feature patches selected by the proposed method, the PAL system calculates the distance between the feature patches of a pixel in the input image and $f \in M$, and then outputs the anomaly score of the pixel.



[Figure 16] Overview structure of proposed model framework

Proposed Method

Irredundant Feature Patch Selection

- Starting from the empty set, the proposed filtering method includes a feature patch sequentially by considering the similarity between the candidate feature patch and already selected feature patches in the set.

- The proposed method includes a new feature patch $f \in F$ added to M at each iteration, where $f = \arg min_{f_i \in F} (1 - \cos(f_i , m_{nn}))$ represents the patch with the smallest Cosine similarity to its nearest-neighbor feature patch in M. $min_{F \in Mc}\ s_i = min_{f \in M_c} \parallel f(x_i) - f \parallel_2$

Algorithm 1 Class-Conditioned Irredundant Feature Patch Selection

Require: Training feature patches $F = \{f_1, f_2, \ldots, f_n\}$ with class labels $C = \{c_1, c_2, \ldots, c_k\}$
Ensure: Heterogeneous memory bank M

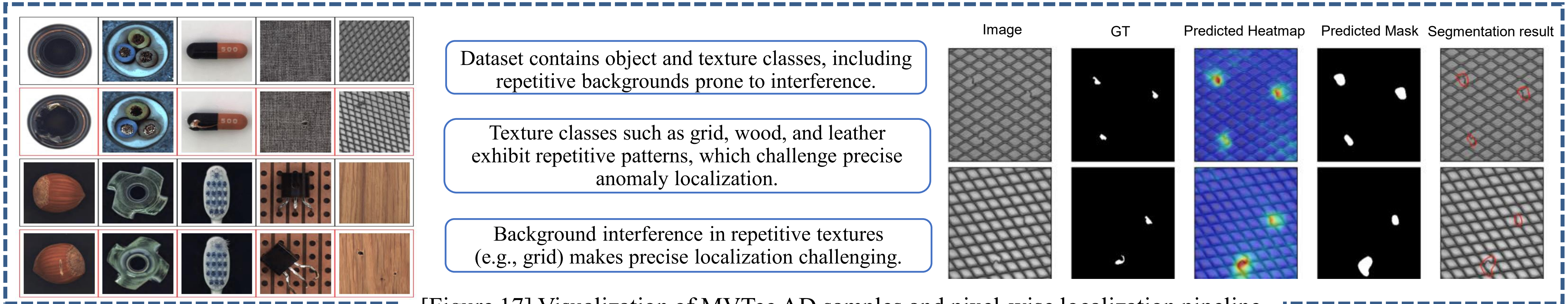
1 $M \leftarrow \varnothing$
2 **for** each class $c \in C$ do **do**
3     $M_c \leftarrow \varnothing$
4   **for** each patch $f_i \in F$ where class$(f_i) = c$ **do**
5   **if** $M_c = \varnothing$ then **then**
6             $M_c \leftarrow M_c \cup \{f_i\}$
7   **continue**
8   **end if**
9   Compute dissimilarity between f_i and selected patches:
    $s(f_i, M_c) = \min_{m \in M_c} (1 - \cos(f_i, m))$
10   **if** $s(f_i, M_c) > \tau$ **then**
11             $M_c \leftarrow M_c \cup \{f_i\}$
12   **else**
13   Skip f_i
14   **end if**
15   **end for**
16     $M \leftarrow M \cup M_c$
17 **end for**
18 **return** M

Line 1 $\rightarrow$ Initialize the global memory M.
Line 2 $\rightarrow$ Iterate over each unique class $c \in \text{set}(C)$.
Line 3 $\rightarrow$ Reset the class-specific memory M_c.
Line 4 $\rightarrow$ Loop over training patches f_i with class$(f_i) = c$.
Line 5–7 $\rightarrow$ If $M_c =$, accept the first patch to seed M_c.
Line 8 $\rightarrow$ Skip scoring for the first patch and continue the loop.
Line 9 $\rightarrow$ Compute dissimilarity $s = \min_{m \in M_c}(1 - \cos(f_i, m))$.
Line 10 $\rightarrow$ If $s > \tau$, the patch is sufficiently novel.
Line 11 $\rightarrow$ Add f_i to M_c to preserve intra-class diversity.
Line 12 $\rightarrow$ Skip f_i as redundant or overlapping.
Line 13 $\rightarrow$ End the conditional branch.
Line 14 $\rightarrow$ Finish the per-class patch loop.
Line 15 $\rightarrow$ Merge class-specific irredundant features into global memory.
Line 16 $\rightarrow$ Merge into global memory $M \leftarrow M \cup M_c$.
Line 17 $\rightarrow$ End the class loop.
Line 18 $\rightarrow$ Return the final heterogeneous memory M.

Experimental Settings

Dataset

Dataset	Train Image	Test Image	Categories	Input Image Size
MVTec-AD	3,629	1,724	15 (5 Textures, 10 Objects)	256×256



[Figure 17] Visualization of MVTec AD samples and pixel-wise localization pipeline

Cross Validation

- Method : Per-class K-Fold on train–non-defective (image-level split; official test fixed)
- Folds × Repeats : 5 × 3
- Data Split Ratio: Train (non-defective) : Validation = 80% : 20% (Test = official fixed split, excluded from CV)

Experimental Settings

[9] Damm, S. et al. “Anomalydino: Boosting patch-based few-shot anomaly detection with DINOv2.” **WACV, 2025.**
[10] Roth, K. et al. “Towards Total Recall in Industrial Anomaly Detection.” **CVPR, 2022**.
[11] Jiang, X. et al. “SoftPatch: Unsupervised Anomaly Detection with Noisy Data.” **NeurIPS, 2022.**
[12] McIntosh, D. and A. Branzan Albu. “Inter-Realization Channels: Unsupervised Anomaly Detection Beyond One-Class Classification.” **ICCV, 2023.**

Hyperparameter Settings

NN = Nearest Neighbor

Method	Batch Size	Epochs	Loss Function	Optimizer	Learning Rate
AnomalyDINO [9] (WACV, 2025)	8 (feature extraction)	0 (index/memory build only)	k-NN distance (L2 or 1−cosine); image score = mean top-1% patch distances	None	(not used)
PatchCore [10] (CVPR, 2022)	8 (feature extraction)	0 (index build only)	k-NN (L2) distance; coreset + FAISS	None	(not used)
SoftPatch [11] (NeurIPS, 2022)	8 (feature extraction)	0 (scoring only)	Distance-based scoring	None	(not used)
InReaCh [12] (ICCV, 2023)	8 (feature extraction)	0 (aggregation only)	k-NN (L2) distance (inter-realization aggregation)	None	(not used)

Evaluation Measures

- Pixel-level Average Precision (AP): AP = $\int_0^1 Precision(Recall)\ dRecall$ Represents the area under the precision-recall curve, evaluating the trade-off between precision and recall for pixel-wise anomaly scores.
- Pixel-level F1 Score: $F_1 = \frac{2\times Precision\times Recall}{Precision\times Recall}$ Represents the harmonic mean of precision and recall, serving as a complementary quantitative metric that balances the trade-off between false positives and false negatives for pixel-wise anomaly detection.

[9] Damm, S. et al. "Anomalydino: Boosting patch-based few-shot anomaly detection with DINOv2." **WACV, 2025.**
[10] Roth, K. et al. "Towards Total Recall in Industrial Anomaly Detection." **CVPR, 2022**.
[11] Jiang, X. et al. "SoftPatch: Unsupervised Anomaly Detection with Noisy Data." **NeurIPS, 2022.**
[12] McIntosh, D. and A. Branzan Albu. "Inter-Realization Channels: Unsupervised Anomaly Detection Beyond One-Class Classification." **ICCV, 2023.**

Performance Comparison

- We conducted a strictly controlled comparative study on MVTec-AD by unifying feature extraction, scoring rules, and memory budget across all methods to isolate the effectiveness of the proposed memory construction strategy.
- In this setting, our method (0.6102) significantly outperformed AnomalyDINO (0.4850) and PatchCore (0.5714), demonstrating that balancing manifold coverage with redundancy reduction is the key factor for maximizing localization accuracy.

[Table 2] A downward arrow indicates statistical significance from paired t-tests against the Proposed method

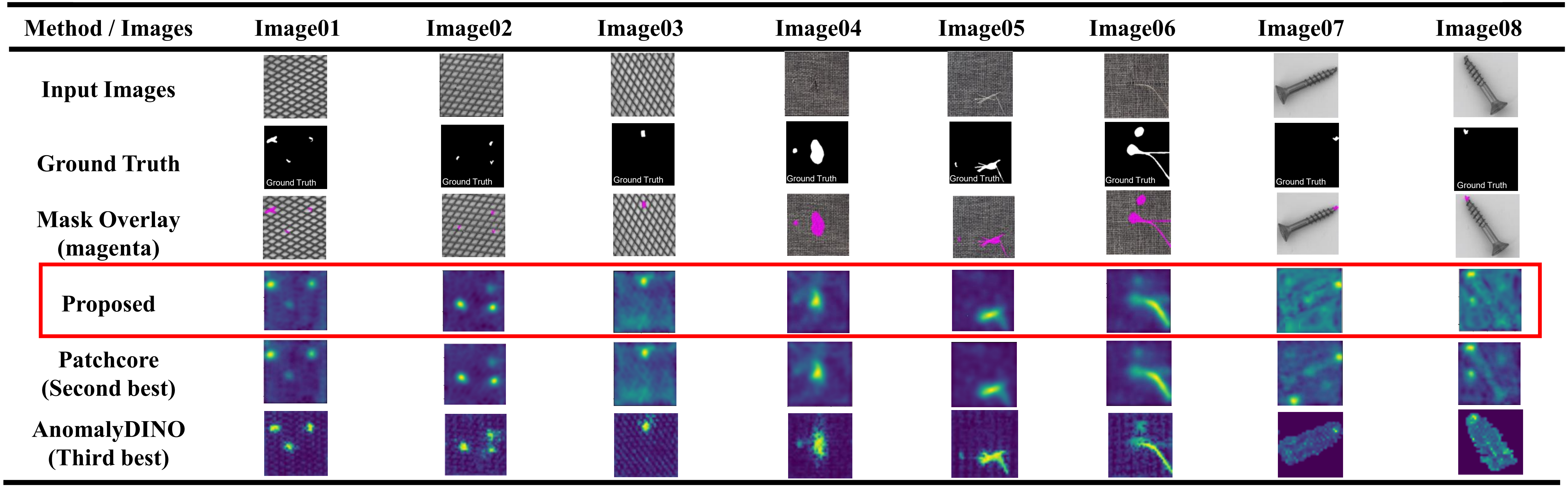
Method	Venue	Pixel AP	F1-score	Precision	Recall
Proposed	**-**	**0.6160 ± 0.1463**	**0.6317 ± 0.1110**	**0.5614 ± 0.1183**	0.7282 ± 0.1069
AnomalyDINO [9]	**WACV, 2025**	0.4850 ± 0.0042	0.5390 ± 0.0028	0.4745 ± 0.0058	0.6382 ± 0.0031
PatchCore [10]	**CVPR, 2022**	0.5714 ± 0.1592	0.5953 ± 0.1300	0.5162 ± 0.1445	0.7158 ± 0.1069
SoftPatch [11]	**NeurIPS, 2022**	0.4055 ± 0.1549	0.4730 ± 0.1321	0.4194 ± 0.1375	0.5771 ± 0.1380
InReaCh [12]	**ICCV, 2023**	0.4810 ± 0.1860	0.1389 ± 0.1120	0.0793 ± 0.0716	**0.9753 ± 0.0319**

Experimental Results

Qualitative Analysis

- We conducted an in-depth qualitative analysis on categories with highly regular patterns to verify that our class-conditioned memory construction effectively suppresses background-like false positives while preserving the precise localization of defect regions.
- The qualitative comparisons demonstrate that while baseline methods often activate on normal repetitive patterns, our method focuses on true defects, producing cleaner anomaly maps and masks that align more closely with the ground truth.

[Figure 18] Qualitative comparison on images with highly regular patterns, showing anomaly maps from methods ranked by pixel AP, including the proposed approach.



Conclusion

- This study presents a patch-based anomaly localization method based on a novel class-conditioned feature patch selection that constructs a memory bank containing irredundant features to improve anomaly localization accuracy.
- The proposed approach outperforms existing methods by enhancing pixel-level detection precision and improving anomaly localization accuracy without requiring additional supervision.
- Experimental results confirm that diversity-preserving feature patch selection reduces redundancy and mislocalization, demonstrating potential for practical industrial inspection systems.

Achievement

- 2017.11 – 2018.10 Head of Inter-University Relations Division, ***46th Student Council, Department of Software, Chung-Ang University***, Seoul, Korea
- Programming Education Volunteer, ***Daehang Elementary School***, Mar 2018 – Apr 2018
- Overseas Field Training Program, ***Chiang Mai University***, ***Thailand*** (***Da Vinci Software Education Institute, Chung-Ang University***), Jan 27 – Feb 9, 2023
- Web Service for Voice Conversion Using Singer’s Vocal Data, ***Chung-Ang University***, 2023
- Real-time Salmon Classification Using Application and YOLO, ***Chung-Ang University***, 2024
- Effective Anomaly Detection by Minimizing Background Interference, ***IEIE***, 2025

Thank you

Reference

[1] Tien, Tran Dinh, et al. "Revisiting reverse distillation for anomaly detection." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), 2023.

[2] A.-T. Ardelean et al., “Blind localization and clustering of anomalies in textures,” in Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR), 2024.

[3] Jin, Ying, et al. "DualAnoDiff: Dual-Interrelated Diffusion Model for Few-Shot Anomaly Image Generation." in Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR), 2025.

[4] Yao, Xincheng, et al. "ResAD: A Simple Framework for Class Generalizable Anomaly Detection." Advances in Neural Information Processing Systems 37(NeurIPS), 2024

[5] Zhang, Zilong, et al. "Industrial anomaly detection with domain shift: A real-world dataset and masked multi-scale reconstruction." Computers in Industry 151 (2023): 103990.

[6] M. Pei et al., “Self-supervised learning for industrial image anomaly detection by simulating anomalous samples,” International Journal of Computational Intelligence Systems, vol.16, no.1, p.152, 2023..

[7] T. Bao et al., “MIAD: A maintenance inspection dataset for unsupervised anomaly detection,” in Proc. of the IEEE/CVF Int. Conf. on Computer Vision (ICCV), 2023.

[8] Simões, Francisco, et al. "Attention modules improve image-level anomaly detection for industrial inspection: A differnet case study." ." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR), 2024

[9] Damm, Simon, et al. "Anomalydino: Boosting patch-based few-shot anomaly detection with dinov2." 2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). IEEE, 2025.

[10] Roth, Karsten, et al. “Towards Total Recall in Industrial Anomaly Detection.” Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022.

[11] Jiang, Xi, et al. “SoftPatch: Unsupervised Anomaly Detection with Noisy Data.” Advances in Neural Information Processing Systems (NeurIPS), 2022.

[12] McIntosh, Declan, and Alexandra Branzan Albu. “Inter-Realization Channels: Unsupervised Anomaly Detection Beyond One-Class Classification.” Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2023.

Future Plan

JOB DESCRIPTION

포지션 (회사)	요구사항
㈜ 한화비전(AI 연구원)	• Machine Learning 기본 지식 보유하신 분 • Deep Learning Framework (TensorFlow, Pytorch 등) 기반 개발 경험 • Detection, Classification, Recognition 등 영상/음원 분석 알고리즘 개발 경험 • (우대) AI 최신 연구에 대한 이해 및 Reproduce 역량 보유 • (우대) NIPS, CVPR, ICML 등 최고 수준 국제학회 논문게재 경험 • 포트폴리오 (PDF) 제출 필수
현대로템 - AI(Model Engineer)	• 학사 이상 • 관련 업무 2년 이상 경험 보유자 • (우대) AI 모델 활용 개발 경험 보유자 • (우대) AI 개발 라이브러리(Anaconda, Pytorch 등) 활용 경험 보유자 • (우대) 소프트웨어 개발 툴(C++,Python 등) 활용 경험 보유자
현대오토에버 [SDx] Backend Developer - 비전 AI 모델 개발	• supervised, semi, unsupervised learning 기반 비전 AI 모델 개발 경험 • 비전 AI 모델의 현장 적용 및 Tunning 경험 • 오픈/상용 영상처리 알고리즘 활용 경험 • classification, detection, segmentation, anomaly detection 모델 최적화

Future Plan

JOB DESCRIPTION

포지션 (회사)	요구사항
롯데이노베이트(Vision AI 기술팀)	• Docker 활용 서비스 개발/ 구축 경험 • Vision AI 관련 전문 지식 : 검출/인식/분류/생성/추적 • Vision AI 관련 프로젝트 경험 • 딥러닝 프레임워크 사용경험: PytorchKeras/Tensorflow 등 • 유관 경력 2년이상 또는 석사 졸업(학사 지원 불가) • (우대)AI Competition 수상자 • (우대)AI 관련 Top-tier 학회 논문 게재 경험 • (우대)AI 모델 Serving 경험 • (우대)석사 학위소지자
하나금융융합기술원(Computer Vision 기술팀)	• (필수) 컴퓨터공학/통계학/수학 관련 학위 • (필수) 머신러닝 이론의 전반적인 지식 보유 • (필수) 딥러닝 프레임워크(TensorFlow, Pytorch 등) 기반 컴퓨터비전 연구개발 경험 • (필수) 프로그래밍(python, c/c++ 등) 능력 • (우대) 컴퓨터 그래픽스 이해/구현/활용 경험 • (우대) AI 관련 최신 논문의 이해/구현/활용 경험 • (우대) 관련 분야 제품 및 서비스 적용 경험 • (우대) 관련 분야 상위급 저널/학회 논문 저자

Future plan

Achievement Plan

Requirement	Achievement plan
프로그래밍 능력 보완	• PCCP 자격증 python 으로 8월 이내에 level 4 까지 수행 • 오픈소스 프로젝트에 기여하거나 개인 프로젝트를 GitHub에 공유하여 프로그래밍 실력을 향상 • 기업별 코딩테스트 및 매주 백준(골드1 달성, 2학차 목표), 프로그래머스 단계별 문제풀이
논문 및 해외 학회 대비 어학능력	• 해외 학회를 대비하여 OPIc 2학차내에 IM3~IH 목표로 취득 • 어학연수(중국) 경험을 바탕으로 HSK 5급~6급 공부(2025년 12월)
프로젝트 관리(포트폴리오)	• 프로젝트를 진행하게 되면, 그 내용을 GitHub에 올려서 꾸준히 commit & push • GitHub에서 오픈소스 AI 프로젝트를 찾고, 이슈 트래커를 통해 버그 리포팅이나 기능 제안 등의 기여 • Kaggle 경진대회 참여하여 최소 2~3(월별)개의 경진대회에 참여하여 모델링 능력을 향상시키고, 리더보드 상위 20% 안에 진입하는 것을 목표(2025년 3~12월)