Patch Memory Bank k-NN for Semi-supervised Visual Hazard Detection in Indoor Mobile Robots

Piotr Wozniak[1][0000−0002−5783−2516] and Tomasz Krzeszowski[1⋆][0000−0001−7359−4637]

Faculty of Electrical and Computer Engineering, Rzeszow University of Technology, al. Powstańców Warszawy 12, 35-959 Rzeszów, Poland
p.wozniak@prz.edu.pl, tkrzeszo@prz.edu.pl

Abstract. This paper presents a semi-supervised patch-memory k-nearest neighbors approach for hazard detection and localization in indoor mobile robots. A convolutional neural network (CNN) backbone is used to extract multi-scale representations, which enable anomaly assessment at both the global image and patch levels via a memory bank of normal patches. Evaluation is performed on a generative benchmark extension of the public MDDRobots dataset, comprising 2,669 test images from independent sequences under varying environmental conditions and covering eight hazard categories. The experimental results show robust performance, with DINOv2 achieving the highest overall performance, reaching image-level ROC AUC = 0.8070 and patch-level PR AUC = 0.8737. The proposed approach is interpretable and suitable for safety-critical robotic perception.

Keywords: visual anomaly detection · hazard detection · semi-supervised learning · k-nearest neighbors · indoor mobile robots

1 Introduction

Ensuring safety in robotics depends on detecting and responding to hazards, which requires recognizing the environment and inferring risks from context. Safety typically comprises two stages: hazard detection and response. The first stage focuses on localizing and identifying risks to enable appropriate reactions [10], a task that is often performed by surveillance robots [3,26]. The second stage involves intervention and an active response built on previously detected hazards [9,23]. These stages are fundamental for protection in complex environments. Advanced applications integrate both, combining continuous safety monitoring with robot-assisted interventions and search and rescue operations [11].

Computer vision plays a critical role in addressing these challenges. Using sensors such as RGB, RGB-D, thermal, or event cameras, robots can determine the location and nature of potential hazards in outdoor [13] and indoor [14,28] environments. These range from minor obstacles that interfere with navigation to

⋆ Corresponding author

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_26

more critical situations that involve dangerous items or human activity. The use of vision in hazard detection is particularly important because it enables robots to capture highly subtle cues present in the scene, which are often crucial for accurately assessing risk. In the literature, various works present different types of hazardous situations, illustrating the diversity and complexity of the scenarios considered [22,39]. Two main approaches to visual hazard detection can be distinguished. The first focuses on recognizing known and predefined hazardous situations, typically detected under supervised or semi-supervised conditions using models trained to classify specific objects or events as dangerous [5], assuming the existence of explicit hazards categories that the robot can identify and respond to. The second approach assumes that danger may manifest as a broadly defined outlier with respect to normal operating conditions, without a clear specification of potentially hazardous objects or situations. In such cases, unsupervised or self-supervised methods play a key role, emphasizing anomaly detection, where detected anomalies are treated as potential hazards [33]. This category places greater emphasis on detecting anomalous patterns rather than identifying specific hazard types, acknowledging that rare or unforeseen dangers in complex environments cannot always be predefined or represented in training data.

For mobile robots, anomaly-based approaches are particularly important, as it is difficult to define all events, changes, or objects that may pose a danger during operation. Visual hazard detection, therefore, often relies on identifying anomalies within the environment, understood as unfamiliar or unexpected parts of the scene. Challenges arise from factors such as lighting variations, changes in equipment, seasonal changes, and the dynamic activity of objects such as humans, which can cause substantial differences between scenes over time. The anomaly detection method must be robust to changes in illumination and camera viewpoint to avoid false positives. At the same time, the sensitivity of the system must be sufficiently high to ensure that potential hazards are not overlooked. This problem is especially critical in unsupervised settings, where no labeled examples of hazards are available, making robust generalization essential [7,41]. Deep learning methods are commonly used for anomaly detection [19,25], but they require large amounts of representative hazard and non-hazard data for proper training, which is difficult to collect. Mobile robots face computational constraints and must perform other tasks, such as navigation and interaction, making hazard detection only one component of a complex system. A robot with prior knowledge of the environment should be able to reliably and efficiently detect various hazards.

A literature review reveals a lack of anomaly detection methods focused on identifying changes and potential hazards in scenes for mobile robots in a dynamic environment. Existing datasets mostly cover static environments, and hazards are often introduced artificially rather than arising from natural changes over time. In this paper, the objects that may pose hazards to robots or humans in indoor environments are introduced and evaluated, addressing this research gap through the following contributions:

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_26

- proposal of a semi-supervised patch memory k-NN method for visual hazard detection with image- and patch-level scoring;
- introduction of a semi-synthetic benchmark created by augmenting real data with hazards generated using generative artificial intelligence;
- provision of an open-source script implementing anomaly detection methods based on three CNN backbones and vision transformer architectures.

2 Related Works

In visual anomaly detection, the goal is to identify deviations from normal patterns that may indicate potential hazards. Detecting anomalies in robotic vision is challenging, as it requires highly stable scene characteristics or models sensitive to contextually significant changes. This is particularly important in robotic scene analysis, where environments can undergo rapid variations due to illumination changes, moving objects, motion blur, and other dynamic factors. A relevant application is hazard detection, which can be treated as visual anomalies requiring timely recognition for safe navigation [21,22]. Notably, the concept of treating anomalies as hazards is not limited to visual methods; non-visual approaches using sensors such as IMU, LiDAR, audio [37], or multi-modal environmental sensors (including temperature, humidity, gas concentrations, air quality, and pressure) can similarly identify deviations indicative of potential danger [15].

Focusing on computer vision, anomaly detection represents a significant challenge due to the diversity of contexts. Over the years, several methodological categories have emerged to address this problem. Image-level methods assess how much an entire image deviates from a normal distribution, producing a single outlier score [18,30]. Patch-level methods analyze local regions individually, estimating the likelihood that each is anomalous[8,38]; notable approaches include PatchCore [27] and PaDiM [8], which leverage patch embeddings to detect fine-grained anomalies efficiently. Pixel-level methods provide dense anomaly maps, highlighting precise regions of deviation [2,4]. In robotics, a balance between detection granularity and processing speed is essential to provide accurate and real-time information.

The detection of anomalies in dynamic indoor environments employs various strategies. Distance-based methods use high-level features from pretrained networks and measure similarity via Euclidean, cosine, or Mahalanobis distances [6,17], flagging regions with high distances, though they can struggle with changing scene contexts and domain shifts. Autoencoders, including convolutional and variational types [40], are trained to reconstruct normal images, where reconstruction errors highlight anomalies at pixel or feature levels. Generative models, such as Generative Adversarial Networks (GANs) and flow-based networks estimate input likelihoods, mark low-likelihood regions as anomalous [16,36]. These are often enhanced with nearest-neighbor searches, local clustering, or top-N patch memory banks to capture subtle or context-dependent deviations.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_26

Self-supervised approaches, such as predicting rotations, patch order, or colorization, identify anomalies when pretext tasks fail, while contrastive learning groups similar observations and separates different ones, flagging features far from known clusters. Additional techniques leverage relational, memory-based [32], and attention-driven models: graph-based methods [1] represent views as object graphs and detect structural deviations, memory-augmented networks store normal representations to identify outliers, and attention or transformer models highlight semantically important regions that conflict with the learned context. Together, these hybrid approaches combine flexibility, interpretability, and robustness, enabling effective detection of rare, fine-grained, and context-dependent hazards in real-world robotic scenarios. Across these methods, leveraging anomaly detection experience allows robots to treat anomalies as hazards, facilitating the identification of semantically inconsistent regions and unexpected dangers in unseen environments.

3 Semi-supervised Patch Memory k-NN Anomaly Detection

This paper proposes a method for visual potential hazard detection in indoor mobile robots. The approach leverages a configurable CNN backbone to extract features and combines global and patch-level analysis using a patch memory bank constructed from normal training images. The method operates without anomaly labels and supports both image-level detection and patch-level localization. The scheme of the method is illustrated in Figure 1.

3.1 Feature Extraction

VGG-16 [31], MobileNetV2 [29], ResNet-18 [12], and DINOv2 (ViT-S/14) [24] serve as pretrained backbone networks for feature extraction. The input images of the network are resized to 224×224 pixels. Feature extraction is divided into patch- and image-level representations to account for the observation that different parts of a backbone provide features of varying quality. Patch-level

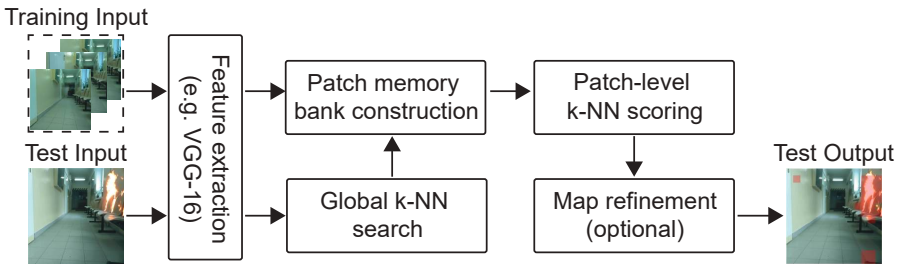

Fig. 1. Overview of the proposed anomaly detection method.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_26

features capture spatial details suitable for anomaly localization, while image-level features provide compact global descriptors for holistic scoring. Not all layers of a CNN are equally informative for localization or global description, motivating the separation into patch- and image-level features. The selection of specific layers for feature extraction is guided by the tensor resolution and channel dimensions to ensure compatibility with the downstream task. In VGG-16, patch-level features are extracted from the convolutional layers up to 'conv5_3' [31], resulting in a feature map of size $512 \times 14 \times 14$, while image-level features are obtained by spatial averaging of the same map, producing a global descriptor of 512 dimensions. In MobileNetV2, patch-level features are extracted from multiple inverted residual stages corresponding to feature maps with spatial resolutions of 56×56, 28×28, 14×14, and 7×7, and channel dimensions of 24, 32, 96, and 320, respectively [29]. These feature maps are projected onto 128 channels, L2-normalized, bilinearly interpolated to a unified 14×14 patch grid, and concatenated to form a $512 \times 14 \times 14$ representation. Channel projection and L2 normalization ensure that all feature maps have a consistent scale and are suitable for comparison in k-NN. Image-level features are derived from the final convolutional output, producing a feature map that is flattened into a 1280-dimensional global descriptor. In ResNet-18, patch-level features are extracted from all four residual stages, projected on 128 channels, normalized, and interpolated to a 14×14 patch grid or the original 224×224 resolution for pixel-level maps, resulting in a patch-level representation of $512 \times 14 \times 14$. Patch-level features in DINOv2 are extracted from the normalized patch tokens of the last transformer layer, capturing both fine-grained spatial details and semantic information. Global descriptors are computed by concatenating three components: mean-pooled features, max-pooled features, and features of the patch that deviates the most from the mean, forming a $3C$-dimensional L2-normalized vector. Patch-level features are obtained via the network's layer-wise spatial outputs, while global features summarize the full image, and both are subsequently used for k-NN-based scoring in anomaly detection.

3.2 Patch Memory Bank Construction

To enable both global and patch-level anomaly scoring, a memory bank of patch features is constructed from normal training images. Each training image is passed through the backbone network to extract spatial features that are then reshaped into individual patch vectors. These patch features are optionally projected to a consistent dimensionality and L2-normalized to ensure comparability. The memory bank also stores global feature representations of each training image, which are used in a preliminary global k-NN search to identify the most semantically similar reference images. Patch-level features are then retrieved from these top-ranked global neighbors for localized anomaly scoring.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_26

3.3 k-NN Global and Patch-level Anomaly Scoring

Anomaly detection is performed using a two-stage k-Nearest Neighbors approach that leverages both global and patch-level feature representations. In the first stage, image-level features extracted from the backbone network are compared with the training set using a k-NN to identify the most visually similar reference images. This global search ensures that the anomaly scoring is guided by semantically relevant images and reduces the influence of unrelated variations. In the second stage, patch-level features from the test image are compared against a patch memory bank constructed from the top-ranked global neighbors. For each patch i, the anomaly score is computed as one minus the average similarity to its k nearest neighbors:

$$s_i = 1 - \frac{1}{k} \sum_{j \in \mathcal{N}_k(i)} \text{cos\_sim}(\mathbf{f}_i, \mathbf{f}_j), \tag{1}$$

where $\mathbf{f}_i$ denotes the feature vector of the test patch, $\mathbf{f}_j$ are the feature vectors of the top-k neighbors from the patch bank, $\mathcal{N}_k(i)$ is the set of indices of these neighbors and cos_sim represents cosine similarity between two vectors. This formulation produces a spatial anomaly map in which higher values indicate a greater deviation from the reference distribution. The combination of global k-NN selection and patch-level scoring allows for accurate and interpretable anomaly localization while maintaining robustness against irrelevant image variations.

3.4 Anomaly Map Refinement

The raw patch-level anomaly map can be optionally refined to enhance spatial coherence and suppress spurious responses. This refinement, referred to as map refinement (MR), smooths the anomaly map to reduce noise and ensure that anomalous regions are more consistent and easier to interpret. In this work, a simple and efficient smoothing operation is applied by convolving the anomaly map with a uniform kernel, effectively averaging local neighborhoods. Specifically, the anomaly map is converted to floating-point format and smoothed using a kernel of size $k_{size} \times k_{size}$ (e.g., 3×3) via a fast blurring operation. This refinement is performed to suppress local noise, aggregate neighboring patch information, and produce more spatially coherent anomaly regions, thereby improving localization accuracy and facilitating interpretation.

3.5 Method Summary

After computing patch-level anomaly scores, the resulting spatial maps are upsampled to the original image resolution to produce pixel-level anomaly masks. These masks allow direct evaluation using the patch-level Precision-Recall Area Under the Curve (PR AUC). The patch-level scores are aggregated across all patches by taking the maximum value to obtain a single image-level anomaly

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_26

score for each test image. This score is used to evaluate global detection performance via the Receiver Operating Characteristic Area Under the Curve (ROC AUC), providing a quantitative assessment of both local and global anomaly detection performance. Compared to PatchCore, proposed method uses a global k-NN search to focus on relevant patches, improving localization and efficiency. Additionally, patch-level scores enable intuitive visualization of anomalous regions by highlighting patches with scores above a chosen threshold. The proposed method combines global and patch-level analysis in a semi-supervised fashion, leveraging a configurable CNN backbone and a patch memory bank constructed from normal training images. The global k-NN search identifies semantically relevant reference images, while patch-level comparisons allow fine-grained localization of anomalous regions. This combination provides robust, interpretable, and scalable anomaly detection suitable for indoor mobile robot applications.

4 Dataset

The dataset was specifically prepared for evaluation purposes, consisting of scenes recorded by a real mobile robot and including natural variability due to environmental changes. To create controlled evaluation scenarios, the dataset was extended with synthetically augmented data. Our approach integrates real-world and synthetically augmented data to enable systematic and robust assessment of anomaly detection performance for mobile robots under diverse conditions. The evaluation dataset and scripts are publicly available at https://doi.org/10.5281/zenodo.19475790, ensuring reproducibility and enabling comparison with future approaches.

4.1 MDDRobots for Hazard Detection

The Multi-Domain Dataset for Robots (MDDRobots) [34] is a benchmark originally designed for indoor Visual Place Recognition (VPR) and anomaly detection (unknown places) in mobile robotics. The dataset comprises 87,750 RGB images captured by multiple robotic platforms under various environmental conditions, including changes in lighting, room layout, and human activity. In this work, MDDRobots is leveraged to address the novel task of visual hazard localization through anomaly detection. The Training sequence and selected samples from Test 1 and Test 3 of the PiCameraRobot subset were employed, representing real-world robotic platforms with different sensing characteristics. The selected test samples were augmented to generate realistic hazards scenarios, introducing eight types of anomalies to allow systematic evaluation. For each image in the Test 1 and Test 3 sequences, defined as normal, anomaly data were generated, resulting in an evaluation set comprising a total of 1,377 images from Test 1 and 1,292 images from Test 3. The original output images produced by the tool had a resolution of 768×576, while the input images were 640×480, requiring normalization of the data to a consistent resolution. This generative approach constructs a tailored benchmark for hazard detection while retaining the diversity and realism of the original dataset.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_26

4.2 Generative Data Augmentation

To generate structured and realistic test data for evaluating the proposed hazard detection method, synthetic data augmentation was employed using a set of carefully designed prompts. Original images, captured by a mobile robot that navigates indoor environments, were processed using the Grok tool [35], which allows controlled modification of visual content based on textual instructions. The corresponding prompts specify the type of hazard to be introduced, such as obstacles, scattered debris, or liquid puddles. The prompts were carefully designed to ensure that the added hazards realistically reflected plausible obstacles in the original scene. While they guided the approximate placement and type of anomaly, they also incorporated a degree of randomness to avoid generating objects in the same or overly predictable locations, ensuring that, for example, an object placed on the floor would not appear in the same spot across images. For synthetic test data, the approach focused on the categories of hazards relevant to indoor mobile robot operation, including static obstacles (box, trash bin), dynamic threats (moving people, fire), slippery substances (liquids), and high-risk items (sharp objects, broken glass). Table 1 presents the prompts used to generate each type of hazard. Each original image is modified according to the selected prompt, creating realistic hazard scenarios that enable systematic evaluation and improve the robustness of the detection model.

Synthetic data is useful for hazard detection, as real examples are often unsafe or hard to capture. Each synthetic image was manually verified (Fig. 2). Global image features were extracted using DINOv2, providing semantically rich representations. Average Euclidean distances indicate minor domain shifts: for Test 1, Train $\rightarrow$ Test Real ≈ 0.3462 and Train $\rightarrow$ Test Synthetic ≈ 0.3548 ($\approx 2.5\%$); for Test 3, Train $\rightarrow$ Test Real ≈ 0.3472 and Train $\rightarrow$ Test Synthetic ≈ 0.3552 ($\approx 2.3\%$), showing very limited differences. Patch-based evaluation and a semi-supervised setup ensure that performance challenges reflect true detection difficulty rather than synthetic bias.

Table 1. Prompts used for generating synthetic visual hazards.

ID	Prompt text	Obstacle type
1	Change the image by placing cables scattered on the floor.	Cables
2	Change the image by adding a trash bin blocking the path.	Trash bin
3	Change the image by adding a box blocking the path.	Box
4	Change the image by placing a trolley blocking the way.	Trolley
5	Change the image by adding a small puddle of liquid on the floor.	Liquid
6	Change the image by adding a person moving unexpectedly.	Person
7	Change the image by adding sharp shards to a part of the floor.	Shards
8	Change the image by setting an existing object in the scene on fire.	Fire

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_26

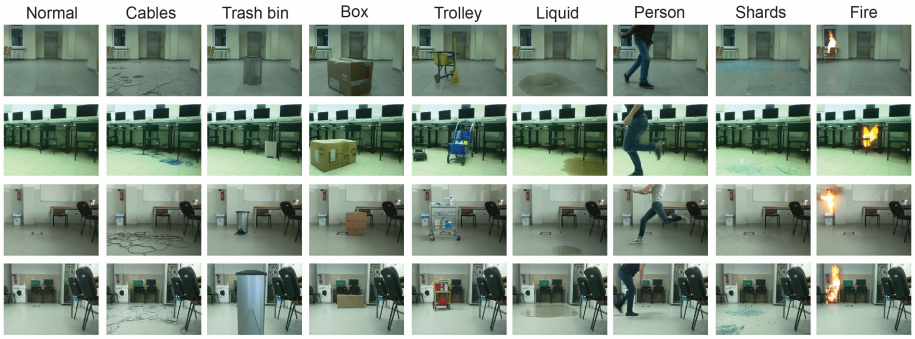

Fig. 2. Example images from the PiCameraRobot subset with generated hazard scenarios.

5 Results

Experiments were conducted at both image and patch levels using nine subclasses: normal, cables, trash bin, box, trolley, liquid, person, shards, and fire. Data were drawn from the Test 1 and Test 3 sequences of the PiCameraRobot subset [34]. The dynamic scenario challenges both detection and localization, and the proposed approach was evaluated under multiple parameter settings and network architectures to assess the impact of feature dimensionality, preprocessing, and model choice on performance. Two evaluation variants were used: a global setting across all classes and a one-class setting comparing anomalies with normal samples for each target class, systematically measuring the robustness and sensitivity of the method.

5.1 Image-level and Patch-level Anomaly Detection

Tables 2 and 3 report one-class anomaly detection results with map refinement for MobileNetV2 (MNet2), ResNet-18, VGG-16, and DINOv2 in RobotPiCamera Test 1 and Test 3, showing the ROC AUC at the image-level and PR AUC at the patch-level. In Test 1, DINOv2 achieves the highest performance both at the image level (ROC AUC = 0.7900) and patch level (PR AUC = 0.8705). Classes such as Person, Cables, and Boxes are reliably detected, whereas smaller or irregular objects (Liquid, Shards, Trash bin) remain challenging. In the more challenging Test 3, DINOv2 again achieves the best ROC AUC at image level (0.8070) and PR AUC at patch-level (0.8332), while VGG-16 maintains steady performance at image-level (ROC AUC = 0.7297) and patch-level (PR AUC = 0.7015). MobileNetV2 shows lower overall performance (ROC AUC = 0.6084, PR AUC = 0.6739). In general, map refinement improves local patch-level sensitivity across all backbones; while DINOv2 excels globally and VGG-16 performs best in select classes, ResNet-18 provides a balanced global performance.

Figure 3 illustrates example query images from the RobotPiCamera Test 1 dataset, overlaid with predicted anomaly masks at a patch resolution of 14×14,

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_26

Table 2. One-class anomaly detection (k = 10) with map refinement (normal vs. anomaly) for RobotPiCamera Test 1

Class	MNetV2+MR		ResNet-18+MR		VGG-16+MR		DINOv2+MR	
	ROC AUC	PR AUC	ROC AUC	PR AUC	ROC AUC	PR AUC	ROC AUC	PR AUC
Cables	0.8000	0.7054	0.8096	0.7349	0.8362	0.7434	0.8669	0.7421
Trash bin	0.6218	0.3679	0.6402	0.3641	0.6755	0.4827	0.6925	0.7452
Box	0.6951	0.5718	0.6501	0.5028	0.7596	0.6651	0.9406	0.8953
Trolley	0.6979	0.6252	0.7042	0.6157	0.7860	0.6632	0.8818	0.8191
Liquid	0.6622	0.4203	0.6062	0.3855	0.7209	0.4553	0.5933	0.4332
Person	0.8364	0.8139	0.8669	0.7994	0.8713	0.8395	0.9588	0.8395
Shards	0.7428	0.6285	0.7470	0.5805	0.7646	0.6341	0.7978	0.6983
Fire	0.7217	0.6438	0.8124	0.6938	0.8189	0.6804	0.5885	0.4971
All	0.7222	0.7383	0.7296	0.7078	0.7791	0.7670	**0.7900**	**0.8705**

Table 3. One-class anomaly detection (k = 10) with map refinement (normal vs. anomaly) for RobotPiCamera Test 3.

Class	MNetV2+MR		ResNet-18+MR		VGG-16+MR		DINOv2+MR	
	ROC AUC	PR AUC	ROC AUC	PR AUC	ROC AUC	PR AUC	ROC AUC	PR AUC
Cables	0.7176	0.6682	0.8058	0.7136	0.8063	0.7114	0.9012	0.7460
Trash bin	0.4655	0.2962	0.5919	0.3242	0.6171	0.3844	0.7368	0.7319
Box	0.5446	0.4590	0.6360	0.4307	0.6889	0.5178	0.9116	0.8482
Trolley	0.5721	0.5159	0.7085	0.5806	0.6978	0.5703	0.8610	0.7652
Liquid	0.5778	0.3495	0.6028	0.2742	0.6794	0.3754	0.6634	0.3646
Person	0.6633	0.7237	0.7709	0.7214	0.8281	0.7686	0.9417	0.8308
Shards	0.7170	0.5958	0.6747	0.4908	0.7620	0.6112	0.8457	0.7044
Fire	0.6094	0.5526	0.7534	0.6486	0.7584	0.5887	0.5944	0.4911
All	0.6084	0.6739	0.6930	0.6511	0.7297	0.7015	**0.8070**	**0.8332**

generated by the DINOv2 model using a patch bank of k=10 nearest neighbors with map refinement. The number of nearest neighbors was empirically determined as a trade-off between robustness and computational cost, using a patch bank of normal training images, whose retrieved neighbors reflect both scene consistency and determine the anomaly score. In the predicted masks, green indicates true positives (TP), red indicates false positives (FP), blue represents false negatives (FN), and the uncolored regions correspond to true negatives (TN). In Figure 3, four representative cases are shown. The first row demonstrates correct anomaly detection with minimal errors, highlighting effective patch bank matching. The second row depicts mislocalized anomalies, where the anomalous regions were not fully captured. In the third row, a large anomalous area is mostly detected; however, a lack of sufficiently similar entries in the patch bank results in incomplete matching. The fourth row presents a FP detection, with normal regions erroneously predicted as anomalous. Some of these errors can be attributed to the influence of anomalous regions in the global feature vectors, which affects the selection of suitable neighbor patches, as well as variations in lighting or object positions in normal scenes that may impact patch-level matching. The proposed solution includes a benchmark mode for reliable anomaly detection,

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_26

controlled by the number of nearest neighbors. In practical robot deployment, it analyzes sequences of images to improve robustness and reduce false predictions.

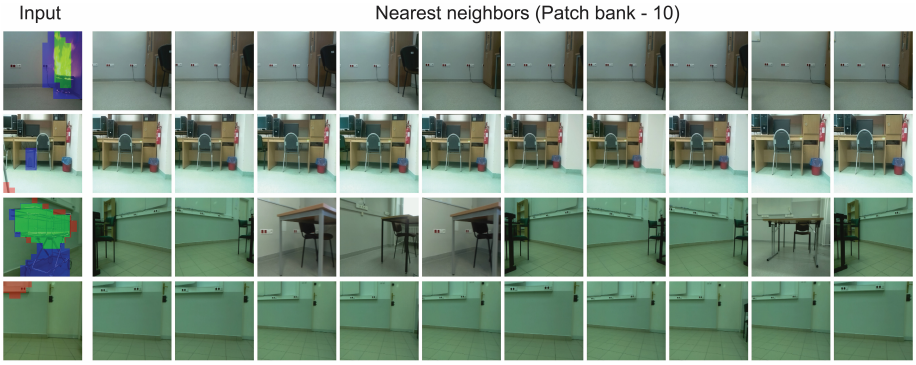

Fig. 3. Query image with predicted anomaly mask and the 10 nearest neighbor images from the patch bank (DINOv2 + MR).

Table 4 summarizes the anomaly detection performance across different backbones (MobileNetV2, ResNet-18, VGG-16, and DINOv2), the number of nearest neighbors (k), and map refinement in RobotPiCamera Test 1 and Test 3. The results show that applying MR and increasing k generally improves the PR AUC at the patch-level, enhancing the sensitivity to local anomalies. DINOv2 consistently achieves the highest image-level ROC AUC (0.8070 in Test 3) and top patch-level PR AUC (0.8737 for Test 1), demonstrating both strong global and local performance. VGG-16 excels in fine-grained patch-level detection (up to 0.7918 in Test 1 with $k = 30$), while ResNet-18 provides strong image-level ROC AUC, particularly with smaller k values. MobileNetV2, though efficient, yields lower overall performance. In general, MR combined with a larger patch bank improves subtle anomaly detection, and the choice of backbone should match the priority: global assessment (DINOv2/ResNet-18) versus local precision (VGG-16).

Feature extraction times were measured on an NVIDIA RTX 3060 Laptop GPU (6 GB). On the training set, DINOv2 averages 10.1 ms per image for global features and 37.6 ms for patch-level extraction, while ResNet-18, VGG-16, and MobileNetV2 require 1.4/28.6 ms, 3.2/27.4 ms, and 1.4/35.3 ms, respectively. This shows that DINOv2 combines high anomaly detection accuracy with competitive efficiency and detailed patch-level representations.

A standard k-NN was used, but inference on resource-constrained robots can be accelerated with FAISS or FAISS HNSW [20]. Using DINOv2 features on Test 1, average search times per image were: standard k-NN ≈ 11.3 ms (global) / ≈ 2.7 ms (patch), FAISS ≈ 1.0 / ≈ 1.1 ms, and FAISS HNSW ≈ 0.5 / ≈ 1.6 ms, showing that approximate methods greatly speed up global search while preserving patch-level accuracy.

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_26

Table 4. Anomaly detection with different backbones, nearest neighbors (k), and map refinement.

Method	Model	#NN (k)	MR	Test 1		Test 3	
				ROC AUC	PR AUC	ROC AUC	PR AUC
PatchCore	PiCamera	–	–	0.5821	0.6755	0.5348	0.6157
MobileNetV2	ImageNet	10	–	0.5309	0.7060	0.5197	0.6375
		10	+	0.7222	0.7383	0.6084	0.6739
		20	+	0.7284	0.7595	0.6273	0.6928
		30	+	0.7461	0.7669	0.6587	0.7013
ResNet-18	ImageNet	10	–	0.7532	0.6825	0.7218	0.6246
		10	+	0.7296	0.7078	0.6930	0.6511
		20	+	0.7559	0.7370	0.7145	0.6743
		30	+	0.7540	0.7511	0.7026	0.6870
VGG-16	ImageNet	10	–	0.5185	0.7040	0.5197	0.6484
		10	+	0.7791	0.7670	0.7297	0.7015
		20	+	0.7918	0.7853	0.7056	0.7151
		30	+	0.7997	0.7918	0.7274	0.7269
DINOv2	LVD-142M	10	–	**0.8070**	0.8184	0.7024	0.7758
		10	+	0.7900	0.8705	**0.8070**	0.8332
		20	+	0.7995	**0.8737**	0.7989	0.8337
		30	+	0.7924	0.8735	0.7917	**0.8362**

6 Conclusions

The proposed approach provides a practical and interpretable framework for detecting potential hazards as anomalies in dynamic indoor scenes. Although the evaluation dataset is designed purely for evaluation purposes and does not contain training data, this aligns naturally with the semi-supervised nature of the method, making it flexible and broadly applicable. A current limitation is the relatively low-resolution ground truth mask, which restricts spatial precision; however, it remains sufficient to identify regions with potential hazards in indoor robotics. Future work could explore higher-resolution mask generation to produce more detailed anomaly maps. In addition, generative approaches could be used to synthesize training data, thereby creating diverse anomalous samples and improving robustness.

The method operates at image and patch levels, localizing unusual regions and providing interpretable visual explanations, though it lacks true pixel-level anomaly detection. The resolution used is adequate for practical robotic hazard detection. The global k-NN search mechanism can also be applied to image retrieval tasks such as VPR, providing a useful extension. Furthermore, the anomaly detection pipeline supports the distillation of a dataset, image comparison, and the detection of outliers as potential new locations. One limitation is the reliance on nearest-neighbor selection for constructing the anomaly bank, which can introduce errors. Additionally, combining the global k-NN search with patch-based bank construction increases computational and time requirements, reducing scalability for very large datasets. Nevertheless, this trade-off is acceptable given the method's interpretability and demonstrated effectiveness in

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_26

controlled evaluation scenarios. Another consideration is that the framework treats all significant deviations as potential hazards without distinguishing true threats from benign variations, such as scene changes caused solely by human activity. Despite this, the approach provides meaningful insight into the dynamics of the environment and can guide safety-oriented decision-making. In general, the semi-supervised, generative-compatible design of the framework, along with its dual-level (image and patch) analysis, makes it a promising foundation for future research on hazard-specific identification and adaptive autonomous robotic perception.

References

1. Akoglu, L., Tong, H., Koutra, D.: Graph based anomaly detection and description: a survey. Data Mining and Knowledge Discovery **29**(3), 626–688 (2015). https://doi.org/10.1007/s10618-014-0365-y
2. Akçay, S., Atapour-Abarghouei, A., Breckon, T.P.: Skip-GANomaly: Skip Connected and Adversarially Trained Encoder-Decoder Anomaly Detection. In: Int. Joint Conf. on Neural Networks (IJCNN). pp. 1–8 (2019). https://doi.org/10.1109/IJCNN.2019.8851808
3. Bao, J., Guo, Y., Song, A., Tang, H.: A multi-agent based robot telesupervision architecture for hazardous materials detection. In: IEEE Int. Conf. on Information and Automation. pp. 2428–2432 (2010). https://doi.org/10.1109/ICINFA.2010.5512282
4. Bergmann, P., Fauser, M., Sattlegger, D., Steger, C.: MVTec AD — A Comprehensive Real-World Dataset for Unsupervised Anomaly Detection. In: IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 9584–9592 (2019). https://doi.org/10.1109/CVPR.2019.00982
5. Celik, T., Demirel, H., Ozkaramanli, H., Uyguroglu, M.: Fire detection using statistical color model in video sequences. Journal of Visual Communication and Image Representation **18**(2), 176–185 (2007). https://doi.org/10.1016/j.jvcir.2006.12.003
6. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection: A survey. ACM Comput. Surv. **41**(3) (2009). https://doi.org/10.1145/1541880.1541882
7. Chen, T., Liu, X., Xia, B., Wang, W., Lai, Y.: Unsupervised Anomaly Detection of Industrial Robots Using Sliding-Window Convolutional Variational Autoencoder. IEEE Access **8**, 47072–47081 (2020). https://doi.org/10.1109/ACCESS.2020.2977892
8. Defard, T., Setkov, A., Loesch, A., Audigier, R.: PaDiM: A Patch Distribution Modeling Framework for Anomaly Detection and Localization. In: Pattern Recognition. ICPR Int. Workshops and Challenges. pp. 475–489. Springer (2021)
9. Edlinger, R., Zauner, G., Zauner, M.: Hazmat label recognition and localization for rescue robots in disaster scenarios. Electronic Imaging **31**(7), 463–1–463–1 (2019). https://doi.org/10.2352/ISSN.2470-1173.2019.7.IRIACV-463
10. Fritsche, P., Zeise, B., Hemme, P., Wagner, B.: Fusion of radar, LiDAR and thermal information for hazard detection in low visibility environments. In: IEEE Int. Symposium on Safety, Security and Rescue Robotics (SSRR). pp. 96–101 (2017). https://doi.org/10.1109/SSRR.2017.8088146

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_26

11. Habib, M.K., Baudoin, Y.: Robot-assisted risky intervention, search, rescue and environmental surveillance. Int. Journal of Advanced Robotic Systems **7**(1), 10 (2010). https://doi.org/10.5772/7249
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep Residual Learning for Image Recognition. In: IEEE Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 770–778 (2016). https://doi.org/10.1109/CVPR.2016.90
13. Hua, M., Nan, Y., Lian, S.: Small Obstacle Avoidance Based on RGB-D Semantic Segmentation. In: IEEE/CVF Int. Conf. on Computer Vision Workshop (ICCVW). pp. 886–894. IEEE Computer Society, Los Alamitos, CA, USA (2019). https://doi.org/10.1109/ICCVW.2019.00117
14. Jeelani, I., Asadi, K., Ramshankar, H., Han, K., Albert, A.: Real-time vision-based worker localization & hazard detection for construction. Automation in Construction **121**, 103448 (2021). https://doi.org/10.1016/j.autcon.2020.103448
15. Khamis, A., Shaban, H.A., Fayed, H.A., Aly, M.H.: Hybrid real-synthetic dataset framework for robotic hazard detection in industrial environments. Scientific Reports **16**(1628) (2026). https://doi.org/10.1038/s41598-025-33603-5
16. Li, H., Li, Y.: Anomaly detection methods based on GAN: a survey. Applied Intelligence **53**(7), 8209–8231 (2023). https://doi.org/10.1007/s10489-022-03905-6
17. Li, Z., Yan, Y., Wang, X., Ge, Y., Meng, L.: A survey of deep learning for industrial visual anomaly detection. Artificial Intelligence Review **58**(9), 279 (2025). https://doi.org/10.1007/s10462-025-11287-7
18. Luo, J., Zhang, J.: A Method for Image Anomaly Detection Based on Distillation and Reconstruction. Sensors **23**(22) (2023). https://doi.org/10.3390/s23229281
19. Ma, X., Wu, J., Xue, S., Yang, J., Zhou, C., Sheng, Q.Z., Xiong, H., Akoglu, L.: A Comprehensive Survey on Graph Anomaly Detection With Deep Learning. IEEE Transactions on Knowledge and Data Engineering **35**(12), 12012–12038 (2023). https://doi.org/10.1109/TKDE.2021.3118815
20. Malkov, Y.A., Yashunin, D.A.: Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. IEEE Trans. Pattern Anal. Mach. Intell. **42**(4), 824–836 (2020). https://doi.org/10.1109/TPAMI.2018.2889473
21. Mantegazza, D., Redondo, C., Espada, F., Gambardella, L.M., Giusti, A., Guzzi, J.: Sensing Anomalies as Potential Hazards: Datasets and Benchmarks. In: Pacheco-Gutierrez, S., Cryer, A., Caliskanelli, I., Tugal, H., Skilton, R. (eds.) Towards Autonomous Robotic Systems. pp. 205–219. Springer (2022)
22. Mantegazza, D., Xhyra, A., Gambardella, L.M., Giusti, A., Guzzi, J.: Hazards&Robots: A dataset for visual anomaly detection in robotics. Data in Brief **48**, 109264 (2023). https://doi.org/10.1016/j.dib.2023.109264
23. Murphy, R.: Human-robot interaction in rescue robotics. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) **34**(2), 138–153 (2004). https://doi.org/10.1109/TSMCC.2004.826267
24. Oquab, M., et al.: DINOv2: Learning robust visual features without supervision. Transactions on Machine Learning Research (2024)
25. Pang, G., Shen, C., Cao, L., Hengel, A.V.D.: Deep Learning for Anomaly Detection: A Review. ACM Comput. Surv. **54**(2) (2021). https://doi.org/10.1145/3439950
26. Paola, D.D., Milella, A., Cicirelli, G., Distante, A.: An Autonomous Mobile Robotic System for Surveillance of Indoor Environments. Int. Journal of Advanced Robotic Systems **7**(1), 8 (2010). https://doi.org/10.5772/7254
27. Roth, K., Pemula, L., Zepeda, J., Schölkopf, B., Brox, T., Gehler, P.: Towards Total Recall in Industrial Anomaly Detection. In: IEEE/CVF Conf. on Computer

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_26

Vision and Pattern Recognition (CVPR). pp. 14298–14308 (2022). https://doi.org/10.1109/CVPR52688.2022.01392
28. Saha, A., Dhara, B., Umer, S., Kulakov, Y., Alanazi, J., Ali, A.: Efficient Obstacle Detection and Tracking Using RGB-D Sensor Data in Dynamic Environments for Robotic Applications. Sensors **22**, 6537 (2022). https://doi.org/10.3390/s22176537
29. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: MobileNetV2: Inverted Residuals and Linear Bottlenecks. In: IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR). pp. 4510–4520. IEEE Computer Society, Los Alamitos, CA, USA (2018). https://doi.org/10.1109/CVPR.2018.00474
30. Vieira e Silva, A.L., Simoes, F., Kowerko, D., Schlosser, T., Battisti, F., Teichrieb, V.: Attention Modules Improve Image-Level Anomaly Detection for Industrial Inspection: A DifferNet Case Study . In: IEEE/CVF Winter Conf. on Applications of Computer Vision (WACV). pp. 8231–8240. IEEE Computer Society, Los Alamitos, CA, USA (2024). https://doi.org/10.1109/WACV57701.2024.00806
31. Simonyan, K., Zisserman, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition. 3rd Int. Conf. on Learning Representations, ICLR 2015 pp. 1–14 (2015)
32. Wang, T., Xu, X., Shen, F., Yang, Y.: A Cognitive Memory-Augmented Network for Visual Anomaly Detection. IEEE/CAA Journal of Automatica Sinica **8**(7), 1296–1307 (2021). https://doi.org/10.1109/JAS.2021.1004045
33. Wellhausen, L., Ranftl, R., Hutter, M.: Safe Robot Navigation Via Multi-Modal Anomaly Detection. IEEE Robotics and Automation Letters **5**(2), 1326–1333 (2020). https://doi.org/10.1109/LRA.2020.2967706
34. Wozniak, P., Krzeszowski, T., Kwolek, B.: Multi-Domain Indoor Dataset for Visual Place Recognition and Anomaly Detection by Mobile Robots. Scientific Data **12**(1), 817 (2025). https://doi.org/10.1038/s41597-025-05124-3
35. xAI: Grok. https://www.grok.com (2025), generative tool for data augmentation. Accessed: 2025-12-04
36. Xia, X., Pan, X., Li, N., He, X., Ma, L., Zhang, X., Ding, N.: GAN-based anomaly detection: A review. Neurocomputing **493**, 497–535 (2022). https://doi.org/10.1016/j.neucom.2021.12.093
37. Yang, Y., Zhao, J., Xu, X., Cao, K., Yuan, S., Xie, L.: Unsupervised Anomaly Detection for Autonomous Robots via Mahalanobis SVDD with Audio-IMU Fusion (2025). https://doi.org/10.48550/arXiv.2505.05811
38. Yi, J., Yoon, S.: Patch SVDD: Patch-Level SVDD for Anomaly Detection and Segmentation. In: Computer Vision – ACCV 2020: 15th Asian Conf. on Computer Vision, Revised Selected Papers, Part VI. pp. 375–390. Springer Int. Publishing, Cham, Switzerland (2021). https://doi.org/10.1007/978-3-030-69544-6_23
39. Yoo, Y., Lee, C.Y., Zhang, B.T.: Multimodal Anomaly Detection based on Deep Auto-Encoder for Object Slip Perception of Mobile Manipulation Robots. In: IEEE Int. Conf. on Robotics and Automation (ICRA). pp. 11443–11449 (2021). https://doi.org/10.1109/ICRA48506.2021.9561586
40. Zhou, C., Paffenroth, R.C.: Anomaly Detection with Robust Deep Autoencoders. In: Proc. of the 23rd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining. p. 665–674. KDD '17, Association for Computing Machinery, New York, NY, USA (2017). https://doi.org/10.1145/3097983.3098052
41. Zoghlami, F., Kurrek, P., Jocas, M., Masala, G., Salehi, V.: Unsupervised Pose Anomaly Detection for Dynamic Robotic Environments. In: IEEE Conf. on Industrial Cyberphysical Systems (ICPS). vol. 1, pp. 338–343 (2020). https://doi.org/10.1109/ICPS48405.2020.9274705

ICCS Camera Ready Version 2026
To cite this paper please use the final published version:
DOI: 10.1007/978-3-032-29918-5_26