Unsupervised Industrial Anomaly Detection via ResNet50 Feature Extraction and k-Nearest Neighbors

Dao Tu Phat
University of Engineering and Technology - Vietnam National University
23020409@vnu.edu.vn

Abstract

Unsupervised anomaly detection plays a crucial role in industrial inspection, especially in scenarios where collecting defective samples is challenging. In this paper, we propose a method that is not only simple and easy to implement but also highly effective. The approach leverages deep feature extraction from a ResNet50 model pretrained on ImageNet. Features from intermediate layers are processed, followed by dimensionality reduction using Principal Component Analysis (PCA), and anomaly scores are computed via the k-nearest neighbors (k-NN) algorithm in the reduced feature space. The proposed method is evaluated on the MVTec AD dataset [2], which contains 5,354 high-resolution images across various object and texture categories with more than 70 types of defects. Experimental results show that this lightweight and interpretable pipeline achieves anomaly detection performance comparable to state-of-the-art methods. Despite its simplicity, the approach delivers strong results, fast inference, and ease of integration, making it a robust and reliable baseline for industrial anomaly detection tasks.

1 Introduction

Unsupervised anomaly detection has emerged as a critical task in computer vision, particularly within industrial inspection. It plays a pivotal role in ensuring product reliability, minimizing errors, and optimizing production costs. A key challenge in this domain is the inherent scarcity of defective samples and the diverse, often unknown, nature of potential anomalies. Unlike supervised learning, which necessitates labeled data for each defect type, unsupervised anomaly detection relies exclusively on normal data during training, subsequently identifying deviations from this learned normality during inference. This characteristic makes it exceptionally well-suited for real-world manufacturing environments, where processes are meticulously optimized to reduce defects, and acquiring comprehensive examples of every conceivable fault type is typically infeasible.

In industrial settings, visual inspection demands the detection of subtle and varied anomalies, ranging from scratches and dents to contaminations or structural deformations. These imperfections can manifest in numerous forms across diverse object types, posing considerable challenges for automated inspection systems. While traditional computer vision techniques often struggle with generalization in such varied contexts, and complex deep generative models typically require extensive training and diverse anomaly datasets, a simpler yet robust approach is often desired.

To address these limitations, in this paper, we propose a simple, easy-to-implement, yet highly effective and interpretable anomaly detection pipeline. Our approach combines deep feature extraction from a powerful pretrained convolutional neural network with a non-parametric anomaly scoring mechanism. Specifically, we leverage ResNet50 - a model extensively pretrained on ImageNet - to extract multi-layer features from input images. We then apply Principal Component Analysis (PCA) for dimensionality reduction, which accelerates inference and reduces noise. Finally, anomaly scores are computed using the k-nearest neighbors (k-NN) algorithm based on the similarity between test features and the distribution of normal training samples. Despite its simplicity, this method demonstrates competitive anomaly detection performance compared to more complex approaches, while offering fast inference and high practical applicability.

We rigorously evaluate the proposed method on the MVTec AD dataset [2], a comprehensive benchmark comprising 5,354 high-resolution images across 15 distinct object and texture categories. Table 1 summarizes the dataset statistics, showing the number of training images, defect-free test images, defective test images, and the number of defect groups per category. The dataset includes pristine, defect-free images for training and images containing 73 diverse types of defects for testing. Figure 1 illustrates example normal and anomalous images from each category, with close-up views highlighting defect regions. This standardized setup facilitates an objective assessment of unsupervised anomaly detection methods in realistic industrial scenarios.

Table 1: Statistical summary of the MVTec AD dataset.

Category	Train	Test (good)	Test (defective)	Defect groups
Carpet	280	28	89	5
Grid	264	21	57	5
Leather	245	32	92	5
Tile	230	33	84	5
Wood	247	19	60	5
Bottle	209	20	63	3
Cable	224	58	92	8
Capsule	219	23	109	5
Hazelnut	391	40	70	4
Metal nut	220	22	93	4
Pill	267	26	141	7
Screw	320	41	119	5
Toothbrush	60	12	30	1
Transistor	213	60	40	4
Zipper	240	32	119	7
Total	3629	467	1258	73

The main contributions of this work are as follows: we propose a lightweight and interpretable anomaly detection pipeline that effectively combines deep feature representations with non-parametric evaluation. We demonstrate the proposed method's competitive performance on a challenging industrial dataset, achieving strong results without relying on complex generative architectures or extensive training. Furthermore, we analyze the model's performance across multiple categories, highlighting its advantages in terms of simplicity, fast inference speed, and reliability.

Through this study, we emphasize the significant potential of pretrained deep features as a strong baseline for industrial anomaly detection, fostering further research towards practical, efficient, and easily deployable methods.

2 Related Work

Anomaly detection is a prominent research area in computer vision, especially in unsupervised settings where only normal data is available during training. As summarized by Bergmann et al. [2], existing approaches can be broadly grouped into four main categories: generative adversarial networks (GANs), deep convolutional autoencoders (CAEs), pretrained deep feature methods, and traditional techniques.

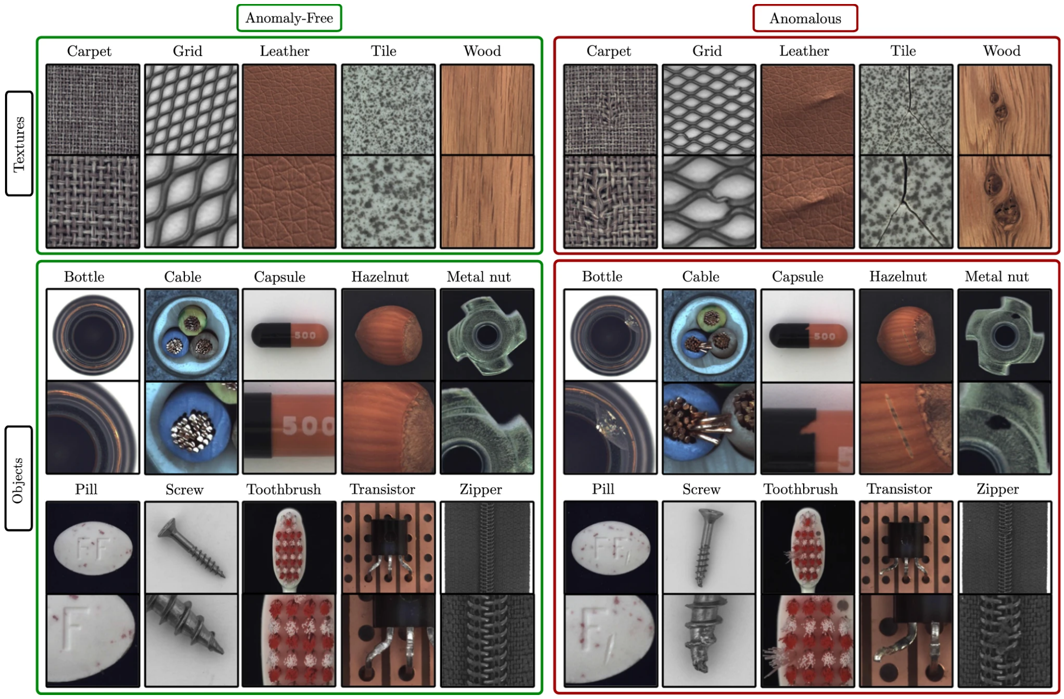

Figure 1: Example normal and anomalous images from all categories in the MVTec AD dataset, with close-up views highlighting defect regions.

2.1 Generative Adversarial Networks (GANs)

GAN-based methods, such as f-AnoGAN [6, 7], aim to model the distribution of normal training data by reconstructing input images and identifying anomalies via pixel-wise reconstruction error. Although effective in capturing coarse anomalies, these methods are computationally expensive at inference time due to the need for iterative optimization. Moreover, they often generate false positives around object edges or specular regions, and may fail to capture fine-grained structural defects due to imperfect reconstructions.

2.2 Deep Convolutional Autoencoders (CAEs)

Autoencoder-based models are widely employed in unsupervised anomaly detection, where the network learns to reconstruct normal inputs, and anomaly scores are computed from reconstruction errors. Bergmann et al. [3] propose leveraging structural similarity (SSIM) to capture local perceptual differences for improved segmentation. Variants such as variational autoencoders (VAEs) and memory-augmented autoencoders have been explored, though they often show limited improvement over standard CAEs. In general, CAEs struggle with reconstructing complex textures or fine color variations, especially in grayscale settings.

2.3 Pretrained Convolutional Features

Recent state-of-the-art methods utilize features extracted from convolutional neural networks pretrained on large-scale datasets such as ImageNet. Napoletano et al. [5] introduced a feature dictionary approach by clustering feature vectors extracted from convolutional networks. Bergmann et al. [1] proposed a student-teacher framework, where student networks are trained to regress features from a frozen teacher network. These methods achieve high-quality, dense anomaly maps and offer strong generalization. However, they may still lack a holistic understanding of global context, leading to missed anomalies that affect overall object structure.

2.4 Traditional Methods

Traditional approaches continue to serve as useful baselines. The GMM-based texture inspection method by Böttger and Ulrich [4] models handcrafted features using Gaussian mixture models, performing well on homogeneous textures. The Variation Model [8] relies on precise alignment of rigid objects and computes reference images using pixel-wise statistics. While effective in controlled environments, such techniques are sensitive to noise and fail to generalize to complex industrial settings.

Summary: Among these categories, approaches leveraging pretrained deep features have demonstrated a compelling balance between accuracy, speed, and robustness. Motivated by this, we propose a simple yet effective pipeline based on deep feature embeddings and non-parametric anomaly scoring for industrial inspection.

3 Method

In this section, we present a simple yet effective anomaly detection method that combines feature extraction from a deep neural network, ResNet50, with the k-nearest neighbors (k-NN) algorithm for anomaly scoring. This method is tailored for detecting localized defects in industrial images by leveraging discriminative and memory-efficient feature representations. The pipeline comprises the following key steps: data preprocessing, multi-layer feature extraction using intermediate layers of ResNet50, patch-level embedding generation, dimensionality reduction using Principal Component Analysis (PCA), construction of a memory bank from normal patches, and anomaly scoring using k-NN. A core advantage of this approach is that it does not require training a new model, uses only normal samples during the training phase, and can be readily applied across various product categories in the MVTec AD dataset.

3.1 Data Preprocessing

We follow the standard MVTec AD dataset protocol, which categorizes high-resolution images into ”good” (normal) and various defective classes. Each image is then resized to 224×224 pixels, normalized using ImageNet statistics, and converted into NumPy arrays for processing by deep convolutional networks.

3.2 Feature Extraction

For feature extraction, we employ a pre-trained ResNet50 model with its classification head removed to access intermediate feature representations. Specifically, we extract feature maps from three layers: `conv2_block3_out`, `conv3_block4_out`, and `conv4_block6_out`. These layers are selected to capture hierarchical information at multiple levels, from low-level textures to higher-level semantics. Each extracted feature map is resized to a fixed spatial resolution of 28×28 pixels using bilinear interpolation to ensure consistency. The resized feature maps are then concatenated along the channel dimension to form a unified, multi-scale feature embedding for each image.

3.3 Patch Embedding and Memory Bank

The concatenated feature maps for each image are reshaped into patch-level embeddings with a shape of $(H \times W, C)$, where H and W represent the fixed spatial dimensions of the resized feature maps. In essence, we convert each spatial location in the feature maps into a compact feature vector, effectively dividing each image into a grid of patches. These patch embeddings from all normal training images are then flattened and combined to construct a global memory bank. This memory bank serves as a reference set that stores typical feature representations from normal data, enabling efficient comparison during the anomaly detection phase.

3.4 Dimensionality Reduction with PCA

To reduce memory usage and computational cost during the nearest neighbor search, we apply Principal Component Analysis (PCA) to the patch-level features from the global memory bank. Specifically, we retain 95% of the total variance, which effectively compresses the feature space while preserving most of its discriminative power. The same PCA transformation, fitted on the normal training data, is then applied to the patch embeddings of the test images to ensure a consistent feature representation across both training and inference phases.

3.5 Anomaly Scoring via Nearest Neighbors

For each patch in a test image, we compute its anomaly score based on the average Euclidean distance to its $k = 1$ nearest neighbor within the global memory bank. A larger distance indicates higher dissimilarity from normal patterns, thus corresponding to a higher anomaly score. These per-patch scores are then reshaped into a spatial map of size $(H \times W)$, effectively localizing potential anomalies within the original image dimensions.

3.6 Evaluation Metric

We evaluate the model's performance using the pixel-level Area Under the Precision-Recall Curve (AUPRC). This metric is preferred as it does not require a fixed threshold for anomaly detection and provides a robust measure of detection quality across various operating points, which is crucial for imbalanced datasets common in anomaly detection. This evaluation method is also adopted in the paper [2].

4 Experiments

We conduct experiments on the MVTec AD dataset [2] using the standard train-test splits. Only normal samples are used during training to build the memory bank, while anomaly scoring is performed on the test set. The model is evaluated using the pixel-level Area Under the Precision-Recall Curve (AUPRC), consistent with prior works. Results are compared against state-of-the-art methods to assess performance.

4.1 Comparison with State-of-the-Art Methods

We compare our approach against several state-of-the-art unsupervised anomaly detection methods previously evaluated on the MVTec AD dataset. Performance metrics for these comparative methods are sourced directly from the original MVTec AD paper [2]. Table 2 summarizes the pixel-level AUPRC results across all 15 MVTec AD categories, including the overall average.

Table 2: Pixel-level AUPRC comparison on the MVTec AD dataset.

Category	f-AnoGAN	ResNet kNN	Feature Dictionary	Student Teacher	ℓ_2-autoencoder	SSIM-autoencoder	Texture Inspection	Variation Model
Carpet	0.025	0.587	0.679	**0.711**	0.042	0.035	0.568	0.017
Grid	0.050	0.335	0.213	**0.512**	0.252	0.081	0.179	0.096
Leather	0.156	0.521	0.276	0.490	0.089	0.037	**0.603**	0.072
Tile	0.093	0.423	0.692	**0.789**	0.093	0.077	0.187	0.218
Wood	0.159	0.468	0.421	**0.617**	0.196	0.086	0.529	0.213
Bottle	0.160	0.773	**0.814**	0.775	0.308	0.309	0.285	0.536
Cable	0.098	0.547	**0.617**	0.592	0.108	0.052	0.102	0.084
Capsule	0.033	**0.450**	0.157	0.377	0.276	0.128	0.071	0.226
Hazelnut	0.526	0.601	0.404	0.585	0.590	0.312	**0.689**	0.485
Metal nut	0.273	0.845	0.760	**0.940**	0.416	0.359	0.153	0.384
Pill	0.121	0.563	0.724	**0.734**	0.255	0.233	0.207	0.274
Screw	0.062	**0.386**	0.017	0.358	0.147	0.050	0.052	0.138
Toothbrush	0.133	**0.585**	0.477	0.567	0.367	0.183	0.140	0.416
Transistor	0.130	**0.443**	0.364	0.346	0.381	0.191	0.108	0.309
Zipper	0.027	**0.643**	0.369	0.588	0.095	0.088	0.611	0.038
Mean	0.136	0.545	0.466	**0.599**	0.241	0.148	0.299	0.234

As shown in Table 2, our ResNet50 k-NN method achieves consistently competitive pixel-level AUPRC results across the MVTec AD dataset categories. While it does not always outperform the state-of-the-art Student Teacher model, our approach demonstrates notably strong performance on categories characterized by subtle texture variations and localized defects, such as *capsule*, *screw*, *toothbrush*, *transistor* and *zipper*, where it achieves the highest AUPRC scores among all compared methods. This highlights the effectiveness of leveraging pre-trained ResNet50 features combined with a simple nearest neighbor search in capturing fine-grained discriminative information necessary for detecting challenging anomalies. However, in categories with more global or semantic anomalies (e.g., *grid*, *leather*, *hazelnut*, *metal nut*), our method performs competitively but slightly below the best-performing methods, indicating potential room for improvement in modeling spatial or contextual cues.

4.2 Analysis of Performance

The strong performance of the proposed method can be attributed to several synergistic factors.

First, leveraging a ResNet50 model pretrained on a large and diverse dataset like ImageNet enables the extraction of highly discriminative features. These features effectively capture fine-grained texture details from earlier layers and high-level semantic information from deeper layers, which is crucial for detecting diverse and subtle anomalies across different scales.

Second, the use of k-NN for anomaly scoring is non-parametric and does not rely on assumptions about the distribution of normal data. This approach directly measures the dissimilarity to the learned normal feature space, enabling effective detection of outliers without requiring complex training on anomalous samples or intricate threshold tuning. This also helps prevent overfitting to limited normal training data.

Finally, compared to complex deep generative models that demand substantial computational resources and elaborate training procedures, our method is simple to implement and fast during inference. It relies solely on normal data to construct the memory bank, making it suitable for real-world industrial applications where anomalous data is scarce or unavailable. Moreover, the pretrained backbone significantly reduces the need for large-scale retraining, enhancing practical deployability.

5 Conclusion

In this work, we have proposed a simple yet highly effective unsupervised anomaly detection method that leverages multi-level deep features extracted from a pretrained ResNet50 backbone combined with a non-parametric k-NN scoring strategy. Our approach requires only normal samples during training and delivers robust, accurate pixel-level anomaly localization on the challenging MVTec AD dataset.

The experimental results demonstrate that despite its conceptual simplicity, the method achieves competitive or superior performance compared to more complex state-of-the-art techniques, especially in identifying subtle geometric and contextual anomalies.

Furthermore, the straightforward design and computational efficiency of the approach make it particularly suitable for real-world industrial inspection scenarios, where anomalous data are scarce or unavailable and minimal parameter tuning is preferable.

Future research could focus on incorporating additional complementary feature representations and adapting the framework to other anomaly detection tasks and domains.

References

[1] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Uninformed students: Student-teacher anomaly detection with discriminative latent embeddings. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4182–4191, 2020.

[2] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. The mvtec anomaly detection dataset: A comprehensive real-world dataset for unsupervised anomaly detection. *International Journal of Computer Vision*, 129:1038–1059, 2021.

[3] Paul Bergmann, Sebastian Löwe, Michael Fauser, David Sattlegger, and Carsten Steger. Improving unsupervised defect segmentation by applying structural similarity to autoencoders. In Alain Tremeau, Giovanni Farinella, and Jose Braz, editors, *14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, volume 5 of *VISAPP*, pages 372–380, Setúbal, 2019. Scitepress.

[4] Tobias Böttger and Markus Ulrich. Real-time texture error detection on textured surfaces with compressed sensing. *Pattern Recognition and Image Analysis*, 26:88–94, 2016.

[5] Paolo Napoletano, Flavio Piccoli, and Raimondo Schettini. Anomaly detection in nanofibrous materials by cnn-based self-similarity. *Sensors*, 18(1):209, 2018.

[6] Thomas Schlegl, Philipp Seeböck, Siegfried Waldstein, Georg Langs, and Ursula Schmidt-Erfurth. f-anogan: Fast unsupervised anomaly detection with generative adversarial networks. *Medical Image Analysis*, 54:30–42, 2019.

[7] Thomas Schlegl, Philipp Seeböck, Siegfried M Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International Conference on Information Processing in Medical Imaging*, pages 146–157. Springer, 2017.

[8] Carsten Steger, Markus Ulrich, and Christian Wiedemann. *Machine vision algorithms and applications*. John Wiley & Sons, 2018.