Contents lists available at ScienceDirect

Current Research in Food Science

journal homepage: www.sciencedirect.com/journal/current-research-in-food-science

A non-destructive, autoencoder-based approach to detecting defects and contamination in reusable food packaging

Anh Minh Truong [*], Hiep Quang Luong

IPI-TELIN, Ghent University-IMEC, Sint-Pietersnieuwstraat 41, Ghent, 9000, East Flanders, Belgium

ARTICLE INFO

Handling Editor: Dr. Maria Corradini

Keywords:
Reusable foodware
Food containers inspection
Artificial intelligence
Auto-encoder
Deep learning

ABSTRACT

Today, environmental sustainability is one of the most critical issue. Hence, the food service industry is actively seeking ways to minimize its ecological footprint. One solution to address this issue is the adoption of reusable foodware in the food service industry. This approach requires a careful process for the collection and thorough cleaning of the foodware, ensuring it can be safely reused. However, reusable foodware might be damaged during the collection process, which can pose food safety hazards for customers. Additionally, there are cases where the cleaning process might not effectively remove all contaminants and therefore cannot be reused after the washing process. To ensure consumer safety, a manual inspection is typically conducted after the cleaning process. However, this step is labor-intensive and prone to human error, particularly as workers' attention may decrease over extended periods. Consequently, the adoption of precise and automated methods for detecting defects and contaminants is becoming crucial, not only to ensure safety but also to achieve scalability and enhance cost-efficiency in the pursuit of environmental sustainability. In our research, we explore various data augmentation strategies and the application of knowledge transfer from various samples of reusable food containers. This method only requires few images from a clean sample to teach the network about normal patterns, and to detect defects by identifying irregular details that do not exist in normal samples. This allows us to rapidly deploy the detection system even with a limited number of collected samples. Experimental results demonstrate the effectiveness of our approach in detecting both contamination and cracks on food containers.

1. Introduction

In recent years, there has been a growing global focus on the importance of reusable foodware and food containers to reduce the use of single-use plastic foodware and minimize plastic waste. Recognizing the urgency of this issue, many countries around the world have set deadlines in the near future for the adoption of reusable food packaging, with the goal of establishing a fully circular packaging system. For example, in Europe, all packaging must be recyclable or reusable by 2030.

In particular, Fevia, the Belgian food industry federation, has set an ambitious target to implement reusable foodware, including containers and cups for food services, five years ahead of the deadline established for all European countries. Since 2020, the use of reusable drinking cups at public events in Belgium has essentially become mandatory. These cups are typically collected and cleaned by social enterprises that specialize in customized work.

The cleaning process often involves several steps to ensure the quality and cleanliness of the cups or containers. Initially, as the cups or containers enter the washing tunnel, they undergo manual inspection. Any items with visible defects are immediately discarded from the process. The cups then progress through four distinct zones within the washing tunnel, each designed to clean them at different temperatures. At the end of the tunnel, there is a final manual inspection to identify any cups that are either defective or not adequately cleaned.

However, this inspection process is time-consuming, subjective, and requires trained personnel. Manual inspection errors typically range between 20% and 30%. Several factors contribute to these mistakes, including exhaustion, stress, solitude, inadequate lighting, loud noises, and inexperience. These human errors can significantly impact overall efficiency, resulting in unnecessary expenses. Additionally, humans cannot efficiently manage the growing inspection demands for large quantities of reusable food packages, underscoring the challenges of manual inspection when dealing with heavy workloads. An automated vision inspection system could enhance the effectiveness of washing cycles (for instance, through pre-sorting or modifying washing settings),

[*] Corresponding author.
E-mail address: anhminh.truong@ugent.be (A.M. Truong).

https://doi.org/10.1016/j.crfs.2024.100758
Received 1 February 2024; Received in revised form 22 April 2024; Accepted 2 May 2024
Available online 13 May 2024

thereby offering substantial benefits in terms of cost savings and environmental impact.

One approach to addressing this task is to train a machine learning model using a supervised learning approach (Saleh et al. (2013)) to detect contaminated and defective areas on reusable foodware. This method involves the collection and labeling of both normal and defective samples. However, this approach faces a significant challenge in collecting a sufficient number of defective samples. While it is easy to gather clean samples, collecting a comprehensive range of defects and contaminants is challenging due to the rarity of defective samples during inspections and the wide variety of potential defects. These defects vary widely in type and severity, including stains, scratches, deformations, and various types of contamination. Consequently, when a new type of anomaly appears during the testing phase, it might not be detected by the classification or segmentation model. Moreover, the number of collected samples can be imbalanced between defective and normal samples, as well as among different classes of defects. This imbalance can result in a bias toward the majority class during the training phase, increasing the likelihood of misclassifying a new sample during testing as belonging to the class with the majority of collected samples. Such behavior is undesirable, as it can lead to the misclassification of food containers.

On the other hand, the problem of identifying defects and contamination in food containers and reusable plastic cups can be considered as an anomaly detection task, where the goal is to detect samples that deviate from the regular distribution. In other words, the objective is to identify all the abnormal or outlier samples that do not fit within the expected patterns (e.g., discoloration, molds, cracks, etc.) of the regular samples. This approach is feasible because manufactured food containers in good condition typically have a high degree of uniformity.

Thus, we can train the network to identify the differences between defected or contaminated food containers and normal ones. Many studies on anomaly detection have shifted in this direction, using the unsupervised learning paradigm (Schlegl et al. (2019); Nalisnick et al. (2019); Collin and De Vleeschouwer (2021)). Generally, the model takes an input image, compresses its information into a latent feature space using an encoder network, and then reconstructs the image from these features with a decoder network. During training, the model is exposed only to defect-free samples. As the encoder compresses the visual information of the input image, the network learns to retain only the essential visual details for effective reconstruction. Consequently, patterns and details from defective samples, which do not appear in the normal samples, are poorly reconstructed. This results in abnormally large reconstruction errors, allowing us to identify defects.

Given that each cup type may feature unique texts, logos, and details, it may be necessary to create multiple models for various cup styles. Therefore, the ability to rapidly deploy the anomaly detection model for new types of reusable plastic cups, even with a limited number of samples, is crucial. To address this problem, we explore different data augmentation strategies as well as strategies for transferring knowledge from the collected samples of different types of reusable cups, aiming to improve the accuracy and robustness of the detection system. In this work, we present experimental results on various types of reusable plastic cups collected from large industrial conveyor dishwashers installed at social enterprises in Flanders, Belgium.

2. Related work

For a significant period, traditional methods have been effectively employed in industrial anomaly detection (Latecki et al. (2007); Steger et al. (2008); Glodek et al. (2013)). Their primary advantage is their low computational cost, which is a crucial requirement for processing the numerous images encountered daily in industrial applications. However, these conventional anomaly detection methods, including the Gaussian Mixture Model-based texture inspection model and the variation model, have limitations in adequately addressing minor defects (e.g., mold spots on containers). They also struggle to accommodate random variations among objects or instances where objects are not perfectly aligned. Consequently, these issues lead to unsatisfactory outcomes.

The limitations of traditional methods have prompted a shift towards more advanced approaches in anomaly detection. With the remarkable success of deep learning across various fields, including computer vision (Krizhevsky et al. (2017); Lin et al. (2018); Ehret et al. (2019); Fernando et al. (2021); Wang et al. (2022)), there has been a significant increase in research interest in applying these techniques to overcome the shortcomings of conventional anomaly detection methods. In recent years, driven by the success of deep learning methods in diverse domains, there has been a notable surge in research interest in the application of deep learning to anomaly detection. This focus is particularly on unsupervised learning for anomaly detection, which can be classified into two different approaches: one utilizing per-pixel reconstruction errors (Seeböck et al. (2020); Collin and De Vleeschouwer (2021); Bergmann et al. (2020)), and the other evaluating the density obtained from the probability distribution of the trained model (Cohen and Hoshen (2020); Gudovskiy et al. (2021); Roth et al. (2022)) in reconstruction-based methods.

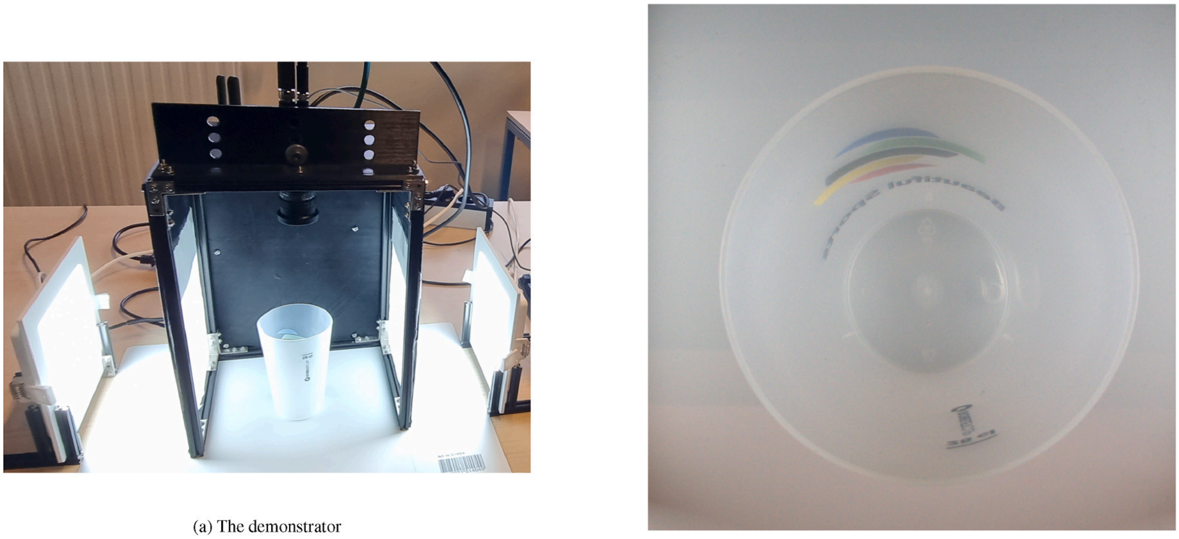
The network, typically based on the architecture of Autoencoders or even Generative Adversarial Networks (GANs), is usually trained using clean images. This training enables the network to learn a compressed representation with a focus on reconstructing normal samples during the training phase. In the testing phase, anomalies can be identified by detecting unusually large reconstruction errors or lower densities in anomalous data. The key idea behind this approach is that the feature compression mechanism in the bottleneck module retains only essential features while disregarding those unseen features associated with anomalies.

However, this is not always true in practice, as demonstrated in Bergmann et al. (2019); Nalisnick et al. (2019). Inaccuracies in reconstructions or poorly calibrated likelihoods can lead to incorrect detections. The feature compression mechanism in the bottleneck module can eliminate the high-frequency information of the input images, resulting in blurry reconstructed images (Collin and De Vleeschouwer (2021)) or a loss of detail. Furthermore, anomaly contamination spots can be extremely small, consisting of only a few pixels, and their textures might not be significantly distinct compared to other parts of the objects, such as labels or patterns on the containers. Consequently, in such cases, the reconstruction errors might not be sufficiently indicative for anomaly detection.

To cope with the diverse sizes and shapes of potential defects, several methods (Gudovskiy et al. (2021); Roth et al. (2022)) have incorporated multi-scale architectures to capture both local and global patch information. Alternatively, the reconstruction problem can be framed as a denoising problem (Mei et al. (2018); Collin and De Vleeschouwer (2021)). During the training process, various types of noise are added to the input image. This approach also allows for the integration of skip connections into the network, preserving high-frequency information and preventing the network from converging solely towards identity mapping.

Recently, diffusion networks and GANs have attracted significant attention due to their performance in data generation. Several works on generating defective samples and using these samples in the training process to improve the accuracy of deep learning models have emerged. For instance, in Karras et al. (2020) and Zhang et al. (2021), these techniques have been applied to generate new defective samples.

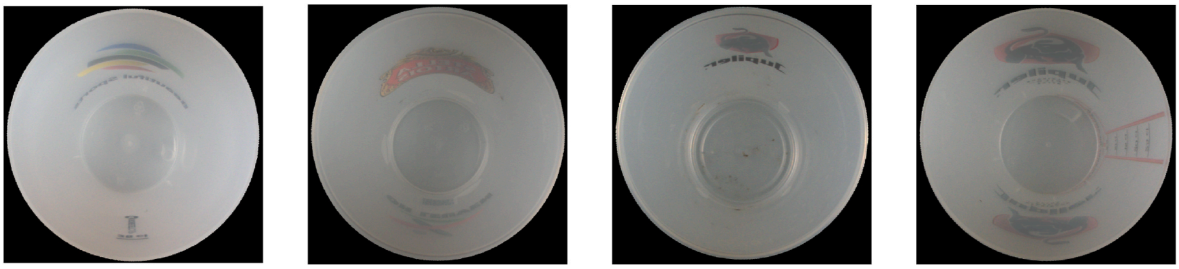
In Hu et al. (2023), the author discusses the fact that previous methods often cannot fill the entire desired input mask with defective details or sometimes produce defects that are not obvious enough. To address this problem, they proposed an Adaptive Attention Weight Map for iteratively enhancing the visual appearance of added defects and filling the entire input anomaly mask. However, this approach is not always beneficial, especially for anomaly detection in plastic containers



(a) The demonstrator

(b) Raw data

Fig. 1. Illustration of the demonstrator built for data capture, along with a display of its raw data outputs.



(a) Cup type 01

(b) Cup type 02

(c) Cup type 03

(d) Cup type 04

Fig. 2. Illustration of the different types of cups: (a) An image of a normal sample from cup type 01. (b) An image of a contaminated sample (mold on the wall of the cup) from cup type 02. (c) An image of a contaminated sample (mold on the bottom of the cup) from cup type 03. (d) An image of a contaminated sample (mold on the wall of the cup) from cup type 04.

and utensils. Since more obvious defects, which can already be easily detected with unsupervised approaches, are less concerning, the focus should be on harder cases, such as minor discoloration or tiny mold spots.

In this study, we focus on exploring various data augmentation and corruption models to enhance the performance of the anomaly detection network for reusable plastic cups. As these cups are mass-produced, the acceptable ones exhibit very similar characteristics with only minor differences, making them often difficult to pinpoint using cameras. Therefore, data augmentation, corruption models, and transferring knowledge can play a crucial role in the training and deployment of the anomaly detection model. In this work, we present experimental results on different types of reusable plastic cups collected from washing companies in Flanders, Belgium.

3. Data acquisition and preprocessing

In this study, we focus on identifying anomalies within reusable plastic cups, capturing a total of 245 images across four distinct cup categories (01, 02, 03, and 04) using the Camera Lucida TRI054S. This camera, equipped with a Sony IMX490 CMOS sensor with a pixel size of 3.0 μm, enables the acquisition of high-resolution images at 2880 × 1860 pixels. We positioned the camera at a height of 30 cm to obtain overhead views of the cups, as depicted on the left side of Fig. 1, and included an example of a raw image capture on the right side of the same figure. For illumination, we installed two 18W LED panels on either side of the demonstrator, approximately 15 cm away—one on the left and one on the right. Each panel measures 220 × 30 *mm* (diameter × height) and features a beam angle of 100°. The panels emit a cool white light (6000K), which resembles natural daylight. Additionally, we placed diffuser paper on the walls of the demonstrator to create diffused lighting, helping to minimize reflections on the cups during image capture.

For image acquisition, the sample was centrally positioned within the frame against a non-reflective white background. The choice of a white background is strategic; it provides a high-contrast backdrop that minimizes interference and simplifies the subsequent background subtraction process. We carefully annotated the areas of anomalies (including cracks, holes, and contamination, among others) present in each image. Some images exhibited multiple types of anomalies.

Cup type 01 featured the largest collection of training samples, totaling 84 images, while the other categories—02, 03, and 04—had considerably fewer training images, with only 5, 5, and 3 images taken from a single sample for cup categories 02, 03, and 04, respectively. This significant difference in sample size poses a distinct challenge for training models to detect anomalies in cup types 02, 03, and 04. Moreover, cup categories 02, 03, and 04 include some samples with patches of light brown or black color, which are very difficult to detect

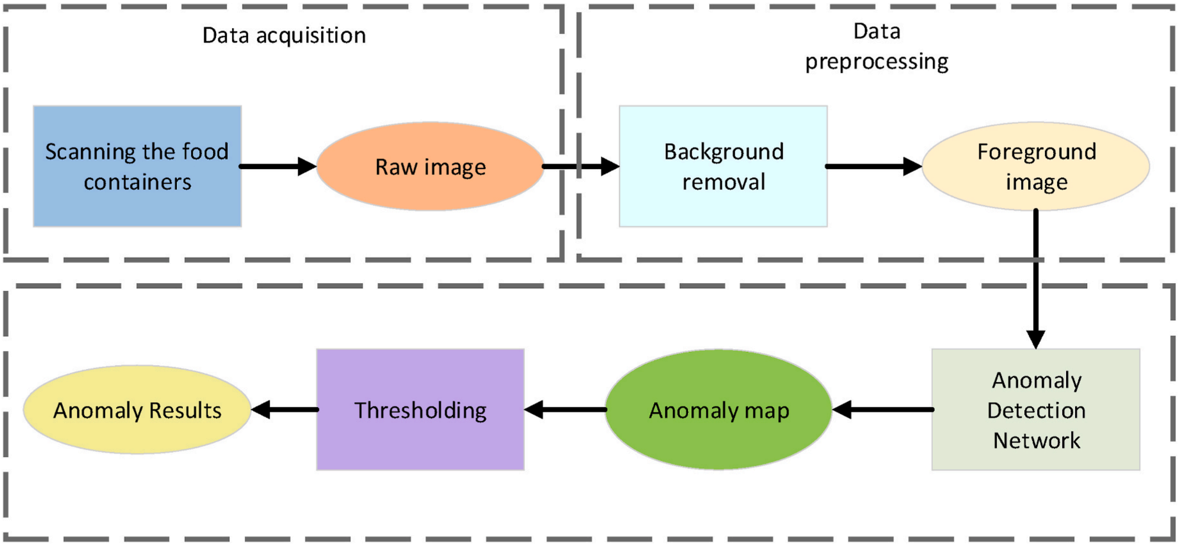


Fig. 3. Overview of the proposed method.

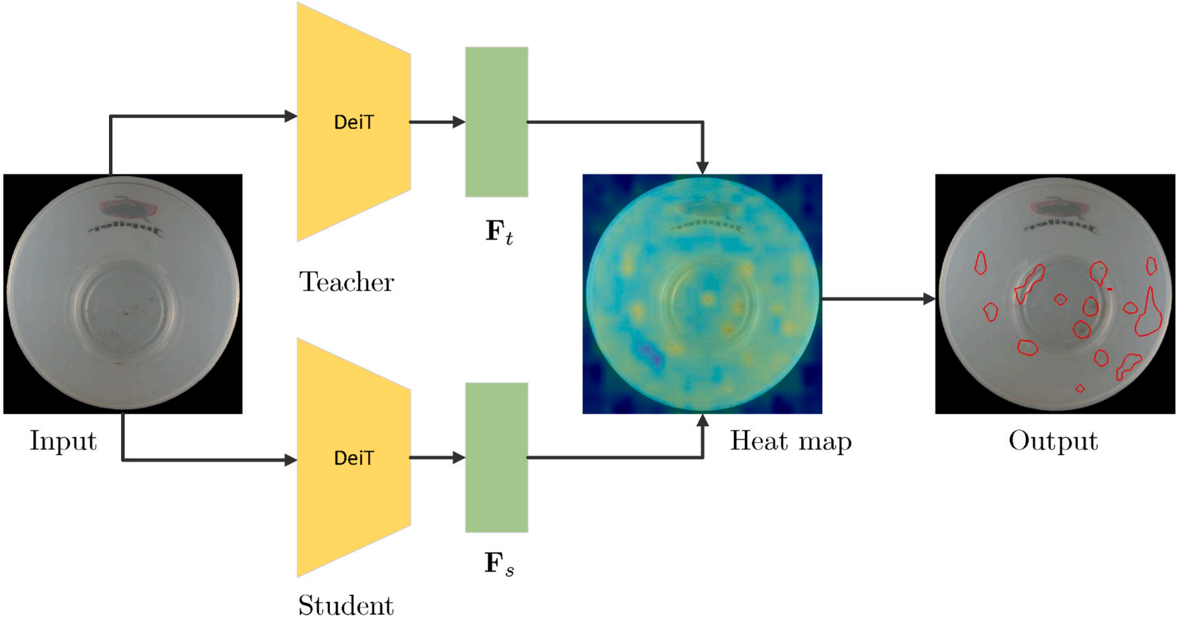


Fig. 4. The architecture of the anomaly detection network.

with the naked eye. This complicates anomaly detection for these types, making the task more challenging compared to type 01, which only contains distinct defects such as mold and cracks.

Prior to processing images through the anomaly detection network, a background removal algorithm based on U-Net is applied to each image. This step ensures that the network's attention is directed towards the cup itself, eliminating distractions from the background. Furthermore, we standardized the orientation of all images within a cup category to ensure consistency, as illustrated in Fig. 2. The images were then resized to 384 × 384 pixels, aligning with the resolution used to train the DeiT network (Touvron et al. (2021)) within our anomaly detection system for optimal feature extraction.

4. Auto-encoder based framework for abnormal detection

In this section, we will discuss the proposed framework of the anomaly detection system for plastic containers. Fig. 3 illustrates the main workflow, which includes the following stages: data acquisition, data preprocessing, and anomaly detection using the proposed autoencoder network. The data were sourced from the data acquisition and preprocessing steps. Subsequently, the proposed autoencoder was employed to determine whether the container is deformed, contaminated, or defect-free. Further details are explained in the subsequent sections.

4.1. Architecture of the anomaly detection network

Fig. 4 depicts the architecture of the anomaly detection network in this work. Let $\mathbf{I}$ represent the input image. We first feed the input image $\mathbf{I}$ to the teacher and student networks to compute the feature maps $\mathbf{F}_t$ and $\mathbf{F}_s$, respectively.

In this work, the anomaly detection network follows a student-teacher scheme, consisting of two networks with identical architectures. However, one is called the teacher network, and the other is called

Table 1
The architecture of the DeiT network.

Layer	Type	Output size
DeiT		
Input	Input Layer	$B \times 3 \times 384 \times 384$
Patch Embed	Patch Embedding	$B \times 576 \times 768$
Token Concat	Concatenation	$B \times 578 \times 768$
Blocks (0–11)	Transformer Block	$B \times 578 \times 768$
Token Remove	Extraction	$B \times 576 \times 768$
Reshape	Reshape	$B \times 768 \times 24 \times 24$
Patch Embedding		
Projection layer	Conv2d	$B \times 768 \times 24 \times 24$
Reshape	Reshape	$B \times 576 \times 768$
Transformer Block		
LayerNorm	Layer Normalization	$B \times 578 \times 768$
Attention	Attention	$B \times 578 \times 768$
MLP	Multi-Layer Perceptron	$B \times 578 \times 768$
Attention		
QKV	Linear	$B \times 578 \times 768$
Proj	Linear	$B \times 578 \times 768$
Multi-Layer Perceptron		
FC1	Linear	$B \times 578 \times 768$
GELU1	GELU	$B \times 578 \times 768$
FC2	Linear	$B \times 578 \times 768$

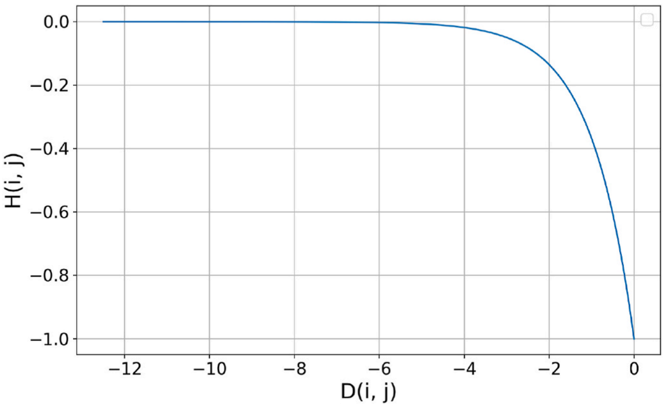


Fig. 5. The illustration of the output range of the $\mathbf{H}(i,j)$.

the student network. The teacher was trained on the ImageNet dataset for image classification, which is very powerful in representing the visual information of the input images. The student network, on the other hand, was trained solely on the defect-free samples that we collected to mimic the feature output of the teacher network. Since the teacher network is trained on a very large and diverse dataset, it can easily represent any visual details of the cup, including defective details, in the deep feature space. On the other hand, as the student network has never seen the defects, it is expected to produce different output features from the teacher network.

Therefore, instead of comparing the reconstruction result of the autoencoder with the input image, we can compare the output feature map $\mathbf{F}_s$ of the student network with the output feature map $\mathbf{F}_t$ of the teacher network. This approach helps mitigate false-positive detections caused by inaccurate reconstructions of normal images (Batzner et al. (2023)).

In this work, we utilized Data-Efficient Image Transformers or DeiT (Touvron et al. (2021)) as the base network for both the student and teacher networks to extract visual features from the input images. DeiT, a vision transformer network, is designed to understand the relationship between global and local features in input data. This enables the network to accurately represent both the shape and the detailed features of reusable cups. This aspect is particularly crucial for anomaly detection performance, as discussed in Yu et al. (2021). The architecture of DeiT is shown in Table 1.

During inference time, we compute the distance map $\mathbf{D}$ between the output feature maps $\mathbf{F}_t$ and $\mathbf{F}_s$ of the teacher and student networks after feature extraction. This computation is shown in Equation 1 as follows:

$$\mathbf{D}(i,j) = \frac{-0.5}{L}\sum_{l=1}^{L}[\mathbf{F}_s(l,i,j) - \mathbf{F}_t(l,i,j)]^2, \quad (1)$$

where $\mathbf{D}(i, j)$ represents the value of the distance map at the spatial location given by the indices i (row) and j (column), and L represents the number of feature channels of the feature maps. The values of $\mathbf{D}(i, j)$ range between $(-\infty, 0]$. If the feature vectors located at (i, j) in $\mathbf{F}_s$ and $\mathbf{F}_t$ are nearly identical, then the value of $\mathbf{D}(i, j)$ will approach zero. Conversely, if they significantly differ, the value of $\mathbf{D}(i, j)$ will tend towards negative infinity. Then, the value of the heat map $\mathbf{H}$ at location (i, j) is computed as shown in Equation (2):

$$\mathbf{H}(i,j) = -\exp[\mathbf{D}(i,j)], \quad (2)$$

In this way, the values of $\mathbf{H}(i,j)$ always fall within a finite range between

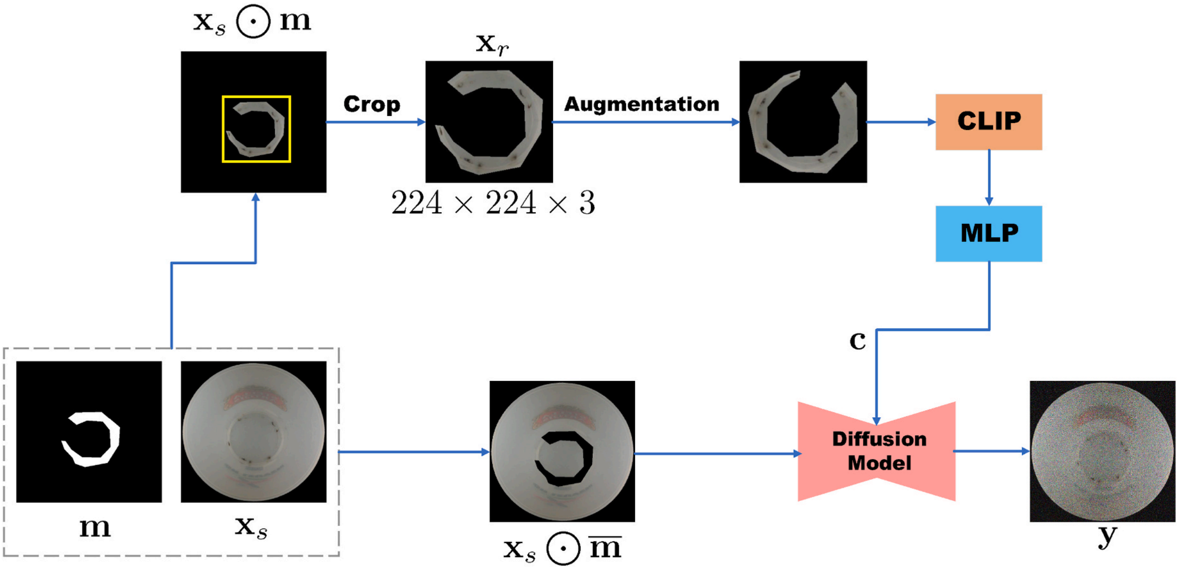


Fig. 6. Illustration of Paint-by-Example training process. Note that ⊙ indicates element-wise multiplication.

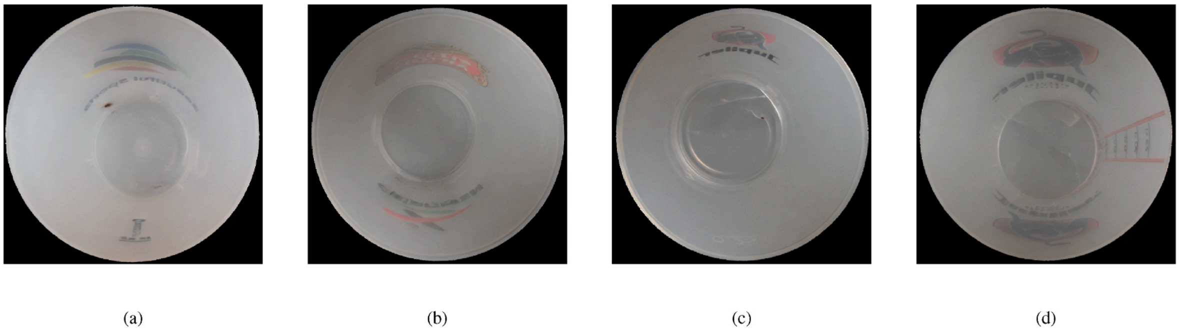


Fig. 7. Illustration of different samples generated by the diffusion model: (a) A generated image of a contaminated sample (mold on the wall of the cup) from cup type 01. (b) A generated image of a contaminated sample (light black area on the bottom of the cup) from cup type 02. (c) A generated image of a broken sample (cracks on the bottom of the cup) from cup type 03. (d) A generated image of a broken sample (cracks on the wall of the cup) from cup type 04.

Table 2
The experimental results on all types of cups are presented below. We highlight the best results with bold text. Higher values indicate better results. Note that 'HF' refers to the hard-case loss function, 'BC' refers to brightness-contrast adjustment, and 'SS' refers to stained-shape noise.

Method	Pixel Metrics		Image Metrics	
	AUROC	F1 Score	AUROC	F1 Score
Baseline				
vanilla	0.913	0.413	0.727	0.830
HF	**0.918**	0.474	0.878	0.958
Data augmentation				
rotation	0.908	0.475	0.881	0.952
BC	0.913	0.459	0.893	0.960
Corruption model				
latent	0.907	0.475	0.902	0.950
SS	0.911	0.467	0.898	0.952
Transfer knowledge				
transfer	0.916	**0.490**	0.895	**0.962**
diffusion	0.915	0.489	**0.919**	0.957

[−1, 0], since $\mathbf{D}(i,j) \in (-\infty, 0]$. $\mathbf{H}(i,j)$ approaches −1 when $\mathbf{D}(i,j)$ approaches 0, and $\mathbf{H}(i,j)$ approaches zero as $\mathbf{D}(i,j)$ approaches negative infinity (as shown in Fig. 5). Similar to Batzner et al. (2023), we then compute the quantile of the heat map as shown in Equation (3):

$$\mathbf{H}_{quantile}(i,j) = \frac{\mathbf{H}(i,j) - q_a}{q_b - q_a}, \tag{3}$$

where q_a and q_b are the normalization parameters. Finally, the anomaly map is computed by thresholding the heat map $\mathbf{H}$. Through quantile-based normalization, the method adapts to the varying distribution of anomaly scores in normal images across different scenarios. Areas with abnormally high values in the quantile heat map $\mathbf{H}_{quantile}$ are classified as anomalies.

4.2. Optimization

During the optimization process, we freeze all the parameters of the teacher network and only optimize the parameters of the student network. In this work, we applied the hard feature loss as proposed in Batzner et al. (2023). The primary objective of this loss function is to backpropagate the loss only from challenging areas. By selectively focusing on these areas, the training concentrates on difficult features that might be misclassified as defects during the testing process. Hence, it can enhance the overall performance of anomaly detection. In Batzner et al. (2023), the author used a method to calculate the hard feature loss based on the squared difference map. However, we have found that we can improve the performance of anomaly detection by using cosine distance or reverse cosine similarity (Deng and Li (2022)) to compute the distance instead.

Table 3
The experimental results for all types of cups are presented below. We highlight the best results with bold text. Higher values indicate better results. Note that 'BC' refers to brightness-contrast adjustment, and 'SS' refers to stained-shape noise.

Method	Pixel Metrics		Image Metrics	
	AUROC	F1 Score	AUROC	F1 Score
State-of-the-art methods				
reverse distillation (Deng and Li (2022))	0.889	0.438	0.391	0.897
efficientAD (Batzner et al. (2023))	0.594	0.129	0.538	0.959
Combination with diffusion				
Diffusion + BC	0.919	0.503	0.937	0.965
Diffusion + Rotate	**0.923**	0.507	0.954	0.961
Diffusion + Rotate + BC	0.920	0.508	**0.957**	**0.969**
Diffusion + Latent + SS	0.919	0.506	0.944	0.965
Diffusion + BC + Latent + SS	0.920	0.501	0.942	0.964
Diffusion + Rotate + Latent + SS	0.917	**0.509**	0.943	0.963
Diffusion + Rotate + BC + Latent + SS	0.917	0.491	0.936	0.957
Combination with transfer noise				
Transfer + BC	0.916	0.476	0.935	0.954
Transfer + Rotate	0.906	0.464	0.933	0.961
Transfer + Rotate + BC	0.917	0.492	0.926	0.961
Transfer + Latent + SS	0.916	0.491	0.938	0.959
Transfer + BC + Latent + SS	0.915	0.482	0.911	0.958
Transfer + Rotate + Latent + SS	0.912	0.482	0.914	0.953
Transfer + Rotate + BC + Latent + SS	0.913	0.479	0.911	0.954
Combination with both diffusion and transfer noise				
Diffusion + Transfer + Rotate + BC	0.914	0.489	0.942	0.953
All	0.921	0.507	0.939	0.959

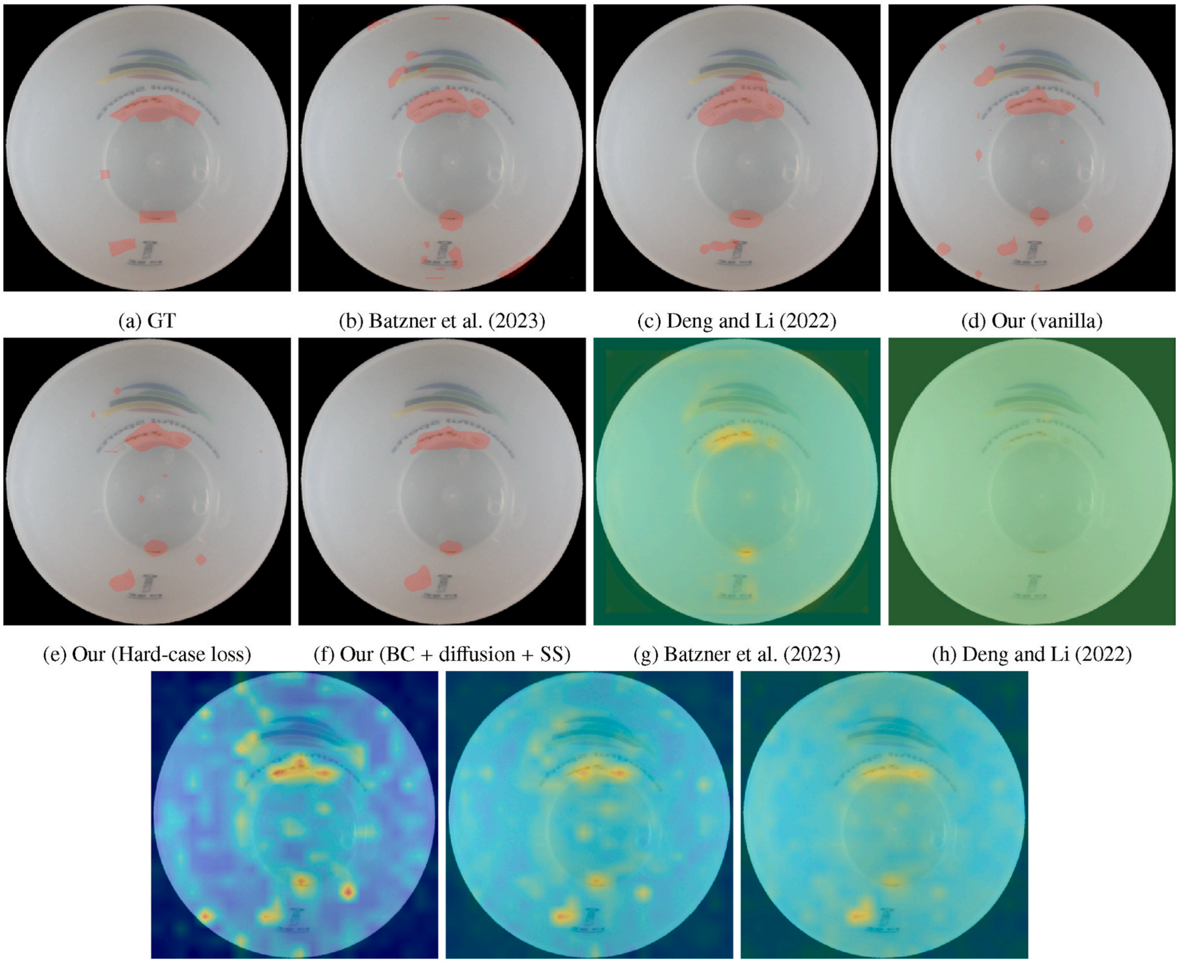
During the training process, we do not compute the heat map as we do during inference time. Instead, we calculate the cosine distance map $\mathbf{D}_{cosine}$ between the feature maps $\mathbf{F}_s$ and $\mathbf{F}_t$, as shown in Equation (4):

$$\mathbf{D}_{cosine}(l,i,j) = 1 - \frac{\mathbf{F}_s(l,i,j) \cdot \mathbf{F}_t(l,i,j)}{\|\mathbf{F}_s(l,i,j)\| \|\mathbf{F}_t(l,i,j)\|}, \tag{4}$$

where $|\cdot|$ represents the Euclidean norm. Subsequently, we determine the threshold d_{hard} based on the mining factor $p_{hard} \in [0, 1]$ such that only $1 - p_{hard}$ percent of the values within $\mathbf{D}_{cosine}$ are greater than or equal to d_{hard}. The hard-case loss L_{hard} is then computed as the average of all elements in $\mathbf{D}_{cosine}$ that are greater than or equal to d_{hard}, as shown below:

$$L_{hard} = \frac{1}{N_{hard}} \sum_{l,i,j} \mathbf{D}_{cosine}(l,i,j) \cdot \tag{5}$$

$$\mathbb{I}(\mathbf{D}_{cosine}(l,i,j) \geq d_{hard}) \tag{6}$$



(a) GT (b) Batzner et al. (2023) (c) Deng and Li (2022) (d) Our (vanilla)

(e) Our (Hard-case loss) (f) Our (BC + diffusion + SS) (g) Batzner et al. (2023) (h) Deng and Li (2022)

(i) Our (vanilla) (j) Our (Hard-case loss) (k) Our (BC + diffusion + SS)

Fig. 8. Examples of segmentation (b–f) and corresponding heatmap (g–k) results for Cup Type 01.

where N_{hard} is the number of elements in $\mathbf{D}_{cosine}$ that are greater than or equal to d_{hard}, and $\mathbb{1}[\cdot]$ is the indicator function, which is 1 when the condition inside the brackets is true and 0 otherwise. In our experiments, we first train the student network using $p_{hard} = 0$ (where all elements are selected) in the first half of the training process. The goal of this phase is to warm up the learning process, and the network already does a decent job of imitating the teacher network. In the second phase, we train the student network using $p_{hard} = 0.99$ (where the top 1% is selected). The goal of this step is to focus on optimizing for the hard features, as mentioned above.

4.3. Data augmentation and corruption model

To further enhance the consistency and accuracy of our anomaly detection network, we have implemented various data augmentation strategies during the training phase. This approach is crucial, considering the limited size and the high similarity of the training samples. Among the augmentation techniques we are exploring, one key strategy is the introduction of rotation to the input images. This adjustment is particularly relevant given that the positioning and orientation of objects, such as cups in photographs, may not always be consistent, leading to slight variations between images.

By randomly rotating the input images by a small degree, we aim to improve the neural network's ability to recognize and correctly classify objects. This process results in a rotated image, denoted as $\mathbf{I}^r$. This image is then fed into both the student and teacher networks, generating corresponding feature maps $\mathbf{F}^r_s$ and $\mathbf{F}^r_t$.

Building on the concept of hard-case loss introduced earlier, we calculate the cosine distance map between these feature maps, $\mathbf{D}^r_{cosine}$, to measure the differences between the features of the rotated image processed by the student and teacher networks:

$$\mathbf{D}^r_{cosine}(l,i,j) = 1 - \frac{\mathbf{F}^r_s(l,i,j)\cdot\mathbf{F}^r_t(l,i,j)}{\|\mathbf{F}^r_s(l,i,j)\|\|\mathbf{F}^r_t(l,i,j)\|}, \tag{7}$$

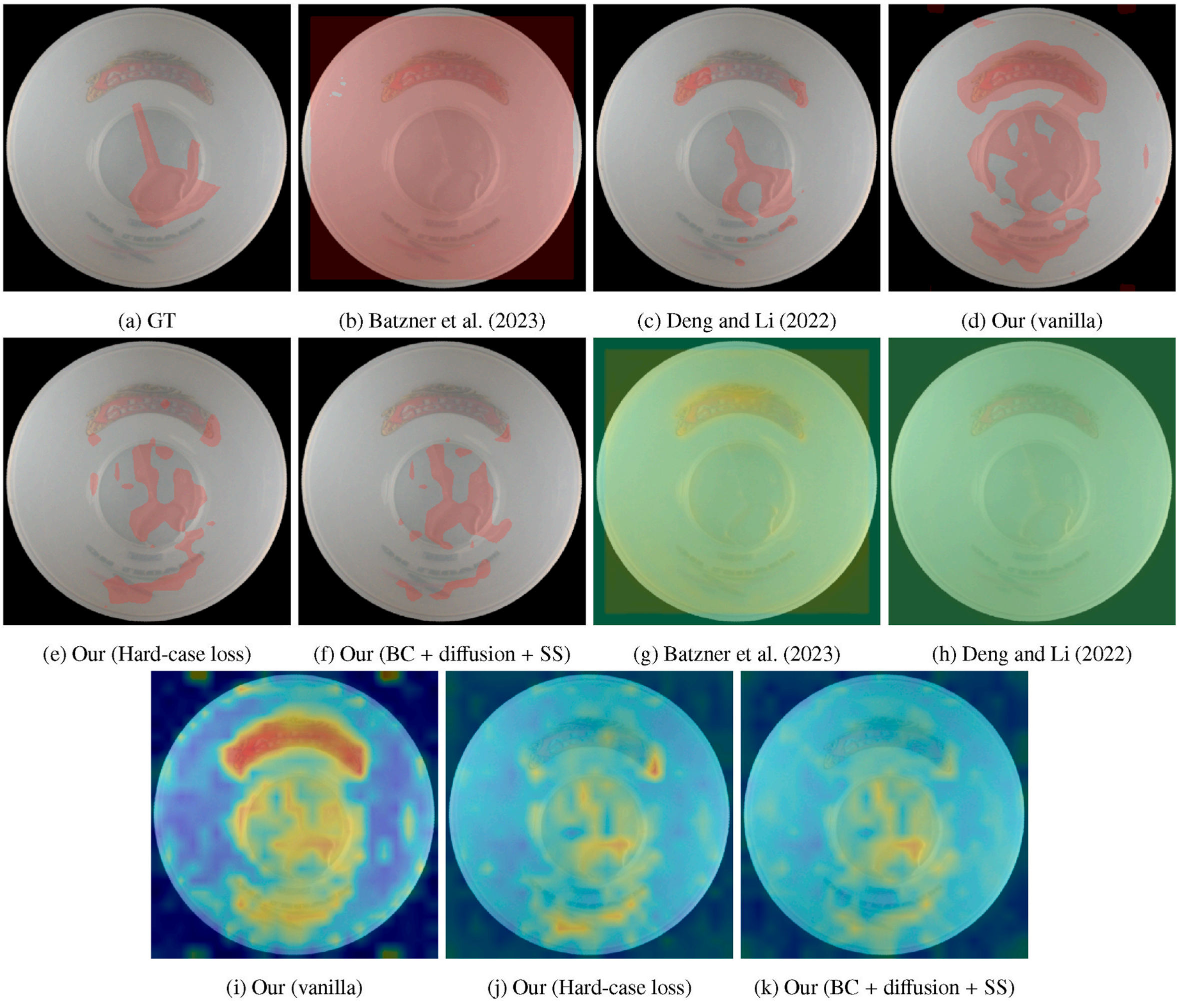
This calculation is similar to Equation (4) mentioned earlier. Finally, we will optimize the student network based on minimizing the hard-case loss L^r_{hard} as below:

$$L^r_{hard} = \frac{1}{N^r_{hard}} \sum_{l,i,j} \mathbf{D}^r_{cosine}(l,i,j)\cdot \tag{8}$$

$$\mathbb{1}(\mathbf{D}^r_{cosine}(l,i,j) \geq d^r_{hard}) \tag{9}$$

where N^r_{hard} represents the number of elements in $\mathbf{D}^r_{cosine}$ that are greater than or equal to d^r_{hard}.

In addition to exploring rotational adjustments, we are also investigating the impact of modifications in brightness and contrast on our anomaly detection network. The rationale behind this exploration stems from the observation that our training samples tend to exhibit a high degree of similarity or homogeneity. Adjusting the brightness and contrast of images is a well-established method for augmenting data in the field of computer vision, aimed at enhancing the robustness and


(a) GT (b) Batzner et al. (2023) (c) Deng and Li (2022) (d) Our (vanilla)

(e) Our (Hard-case loss) (f) Our (BC + diffusion + SS) (g) Batzner et al. (2023) (h) Deng and Li (2022)

(i) Our (vanilla) (j) Our (Hard-case loss) (k) Our (BC + diffusion + SS)

Fig. 9. Examples of segmentation (b–f) and corresponding heatmap (g–k) results for Cup Type 02.

accuracy of neural networks.

By randomly adjusting the brightness and contrast levels of our training images, we introduce a broader spectrum of visual variations. This augmentation strategy enriches the collected dataset, providing the neural network with a more comprehensive learning experience. We expect that the network's ability to generalize across different lighting and visual conditions is significantly improved, leading to enhanced performance.

Similar to random rotation approach, we adjust an image's brightness and contrast to produce a modified version, denoted as $\mathbf{I}^{bc}$. This image is then processed by both the student and teacher networks, resulting in the generation of feature maps $\mathbf{F}_s^{bc}$ and $\mathbf{F}_t^{bc}$, respectively. Then, we can optimize the student network's parameters by minimizing the hard-case loss L_{hard}^{bc}, which is calculated based on the cosine distance map between $\mathbf{F}_s^{bc}$ and $\mathbf{F}_t^{bc}$. The loss function is formulated as follows:

$$L_{hard}^{bc} = \frac{1}{N_{hard}^{bc}} \sum_{l,i,j} \mathbf{D}_{cosine}^{bc}(l,i,j) \cdot \tag{10}$$

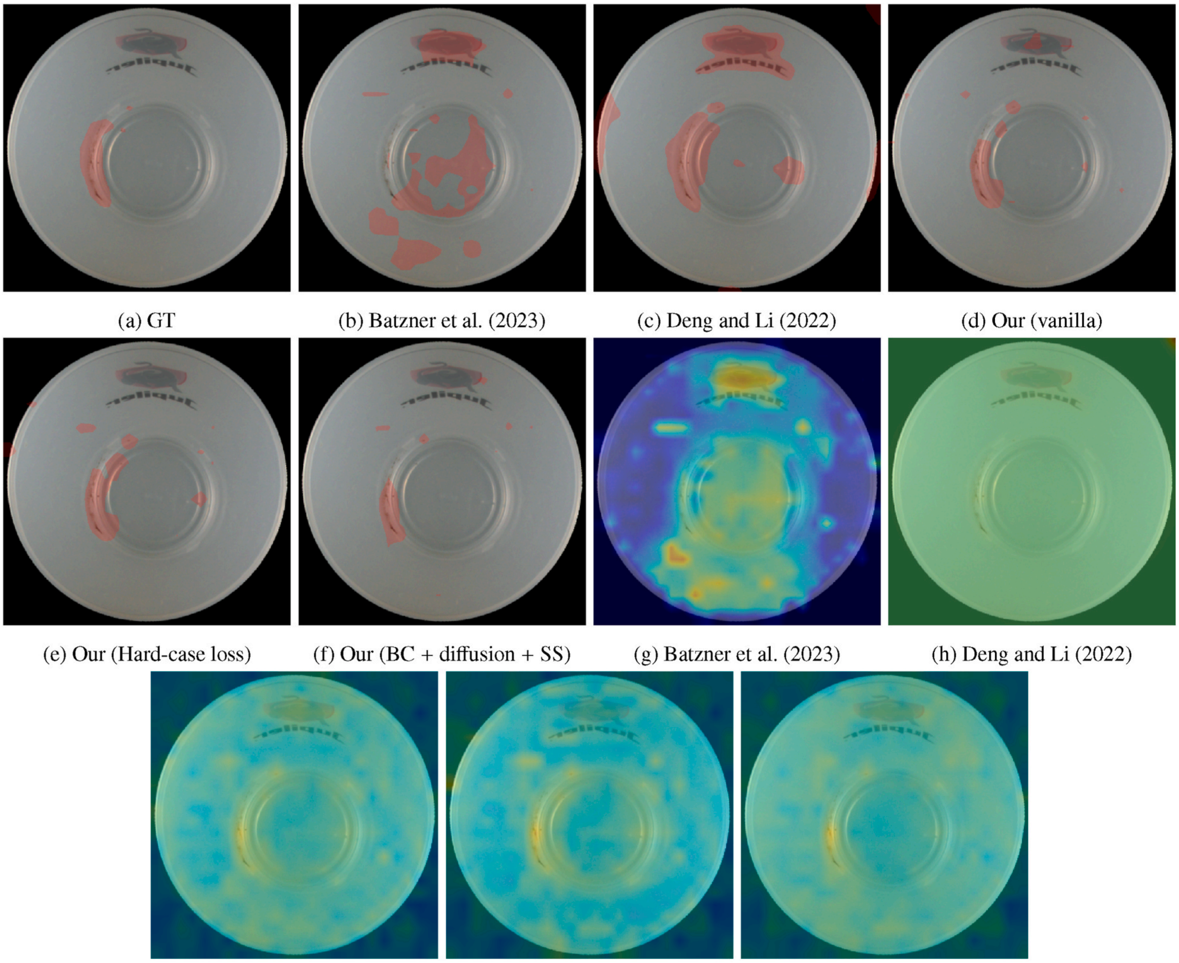
$$\mathbb{1}(\mathbf{D}_{cosine}^{bc}(l,i,j) \geq d_{hard}^{bc}) \tag{11}$$

In this equation, N_{hard}^{bc} represents the number of elements in the cosine distance map $\mathbf{D}_{cosine}^{bc}$ that meet or exceed a predefined threshold d_{hard}^{bc}. Through this approach, we aim to mitigate the network's sensitivity to subtle variations in brightness and contrast, thereby enhancing its overall detection capabilities.

Alongside rotation and brightness-contrast adjustments, we have evaluated different corruption model techniques to improve the performance of anomaly detection models. One state-of-the-art approach is the stained-shape noise technique, as proposed in Collin and De Vleeschouwer (2021). This approach introduces a novel form of data corruption by overlaying the input image, denoted as $\mathbf{I}$, with irregularly shaped ellipsoids that vary in color, thereby creating a corrupted version of the image, represented as $\mathbf{I}^{ss}$.

The training process of the stained-shape noise involves feeding this corrupted image $\mathbf{I}^{ss}$ into the student network, which then generates a specific feature map, $\mathbf{F}_s^{ss}$. Simultaneously, the teacher network still processes the original image $\mathbf{I}$ to produce feature map $\mathbf{F}_t$. The next step focuses on refining the student network by minimizing the hard-case loss L_{hard}^{ss}, which is calculated based on the cosine distance map between $\mathbf{F}_s^{ss}$ and $\mathbf{F}_t$. This calculated loss guides the student network to adjust its outputs to closely match those derived from clean, uncorrupted images. Thus, this makes the differences of defective details in the feature space between $\mathbf{F}_s^{ss}$ and $\mathbf{F}_t$ more pronounced. Consequently, the anomaly detection model becomes more adept at identifying deviations from the norm.

Expanding on our investigation into corruption models, we explore the feasibility of introducing Gaussian noise into the feature space. Inspired by SimpleNet (Liu et al. (2023)), our approach involves adding Gaussian noise directly to the output feature map of the initial patch embedding layer of the DeiT architecture within the student network. The addition of Gaussian noise strategically disturbs the feature representations extracted from the input image, embedding irregular patterns



(a) GT (b) Batzner et al. (2023) (c) Deng and Li (2022) (d) Our (vanilla)

(e) Our (Hard-case loss) (f) Our (BC + diffusion + SS) (g) Batzner et al. (2023) (h) Deng and Li (2022)

(i) Our (vanilla) (j) Our (Hard-case loss) (k) Our (BC + diffusion + SS)

Fig. 10. Examples of segmentation (b–f) and corresponding heatmap (g–k) results for Cup Type 03.

within these features. This presents a challenge for the downstream layers of the network, pushing them to adapt and learn how to refine their output features to align with the characteristics of clean, undisturbed samples.

In this work, we refer to this intervention as the "latent noise" technique. Similar to the training process of stained-shape noise, the output of the student network after this process is denoted as $\mathbf{F}_s^l$. We also optimize the student network by minimizing the hard-case loss L_{hard}^l on the cosine distance map between $\mathbf{F}_s^l$ and $\mathbf{F}_t$.

4.4. *Knowledge transfer from previously collected samples of other cup types by data augmentation*

The goal of the corruption models we've discussed is to manipulate the distribution of visual features extracted from any input sample, aligning it more closely with the distribution seen in defect-free samples. This alignment is intended to increase the distance between the features produced by the student and teacher networks. However, an important consideration is that the alterations made to the images or feature maps in the previously mentioned techniques lack realism. Consequently, these modifications might not effectively represent real-world defects or contamination. Thus, this might introduce unwanted bias to the anomaly detection network.

In this section, we shift our focus to a different set of strategies. Our goal is to transfer the knowledge about defects and contamination, which we have previously gathered from various types of reusable cups, to a new type of cup. This transfer of knowledge is crucial; it allows us to apply the principles of the corruption model in a way that more accurately reflects actual anomalies. By doing so, we can significantly enhance the precision and reliability of our anomaly detection model.

Expanding upon the concept of latent noise, we have also incorporated a technique called transfer noise. This approach involves adding noise to the output feature map of the patch embedding layer within the student network. However, in contrast to latent noise, transfer noise specifically utilizes noise patterns derived from defective samples across various cup types. This technique is grounded in the realization that the previously discussed stained-shape and latent noises fall short of accurately mimicking the real defects found on reusable cups. Our goal with transfer noise is to enhance the model's robustness with the characteristics of real defects, thereby improving its ability to recognize such anomalies in new types of cups.

The procedure begins by selecting a random pair consisting of a defective sample, denoted as $\mathbf{I}^{dt}$, and a normal sample, $\mathbf{I}^{nt}$, each from different cup varieties. These samples are fed into the teacher network to extract their respective feature maps from the patch embedding layer of the DeiT model, resulting in $\mathbf{F}_{et}^{dt}$ for the defective sample and $\mathbf{F}_{et}^{nt}$ for the normal sample. Following this, we compute a differential map, $\mathbf{M}^{transfer}$, which captures the distinctions between these feature maps, as follows:

$$\mathbf{M}^{transfer} = \mathbf{F}_{et}^{dt} - \mathbf{F}_{et}^{nt}. \tag{12}$$

Then, this differential map is added to the feature map $\mathbf{F}_{es}$ produced by the patch embedding layer of the student network to create a new patch-

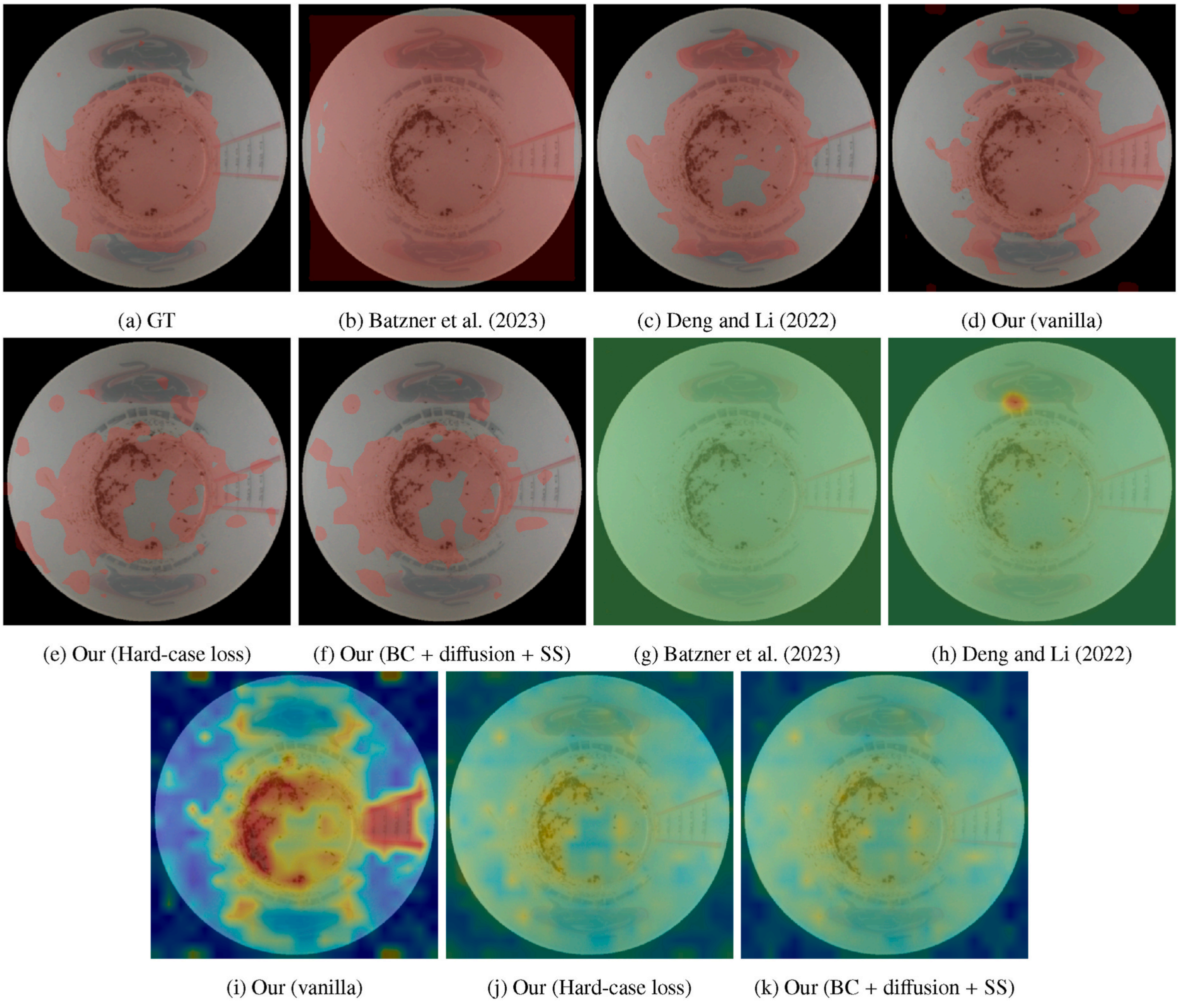

(a) GT
(b) Batzner et al. (2023)
(c) Deng and Li (2022)
(d) Our (vanilla)
(e) Our (Hard-case loss)
(f) Our (BC + diffusion + SS)
(g) Batzner et al. (2023)
(h) Deng and Li (2022)
(i) Our (vanilla)
(j) Our (Hard-case loss)
(k) Our (BC + diffusion + SS)

Fig. 11. Examples of segmentation (b–f) and corresponding heatmap (g–k) results for Cup Type 04.

embedded feature map $\mathbf{F}_{es}^{transfer}$, as shown in Equation (13).

$$\mathbf{F}_{es}^{transfer} = \mathbf{F}_{es} + \mathbf{M}^{transfer}. \tag{13}$$

This new embedded feature map is then fed into the remainder of the DeiT network within the student network to produce $\mathbf{F}_{s}^{transfer}$. Finally, we optimize the student network by minimizing the hard-case loss $L_{hard}^{transfer}$ based on the cosine distance map between $\mathbf{F}_{s}^{transfer}$ and $\mathbf{F}_{t}$.

Diffusion models have recently emerged as a powerful tool for data generation, gaining significant interest in various applications. In our research, we utilized the "Paint-by-Example" diffusion architecture (Yang et al. (2023)) to transfer defect details from one reusable plastic cup sample to another. Similar to stained-shape noise, we also input the generated synthetic image to the student network and train it to mimic the output features of the original image. The goal is for the student network to replicate the original image's features, learning to isolate and eliminate defects from the feature space. Consequently, this process amplifies the feature output disparities between the student and teacher networks in defective regions, enhancing the model's defect detection capabilities.

This process is similar to the scenario where we aim to retrain the model on a new type of cup, utilizing the collected data from other types of reusable cups to improve the performance of the detection network and expedite the deployment of a new model. In this work, we selected the 'Paint by Example' diffusion architecture (Yang et al. (2023)) because it enables us to target specific types of defects or contaminations that have not been effectively detected. This is achieved by generating additional synthetic samples that closely resemble these defects.

Given that the diffusion network was initially trained on a dataset vastly different from our requirements, it necessitated retraining with our collection of defective samples. This retraining process is depicted in Fig. 6. Let an image from a defective sample is represented as $\mathbf{x}_s \in \mathbb{R}^{H\times W\times 3}$, where H and W represent the image's height and width, respectively. $\mathbf{m}$ is a binary map distinguishing defects/contaminants (indicated by 1) from normal pixels (indicated by 0). A reference image, $\mathbf{x}_r$, is then created by cropping and resizing the defective area to a 224 × 224 resolution. The diffusion network proceeds to synthesize an image $\mathbf{y}$ from the set $\{\mathbf{x}_s, \mathbf{x}_r, \mathbf{m}\}$. The defective patterns within $\mathbf{x}_r$ undergo compression into a one-dimensional vector, denoted as $\mathbf{c}$ (with 1024 channels), through the use of the image CLIP network and the MLP. This compression process retains semantic information while disregarding high-frequency details. This step ensures the network to understand the semantic information of the defects and prevents the generator from simply copying and pasting to attain optimal training results. Consequently, this procedure ensures that the synthesized normal regions closely resemble the original source image $\mathbf{x}_s$, while the defective areas are inpainted to align with the defect patterns observed in the reference image $\mathbf{x}_r$.

Instead of retraining the entire model, we applied Low-Rank Adaptation (Hu et al. (2022)) to introduce trainable parameters into the frozen pretrained diffusion network. This approach significantly reduced the total number of trainable parameters, greatly decreasing the storage requirements for the training process. This allowed us to quickly adapt the network even with a small number of samples. Examples of the generated images are illustrated in Fig. 7. The generated image $\mathbf{I}^{diff}$ is then input into the student network to create the corresponding feature map $\mathbf{F}_s^{diff}$. We then optimize the student network by minimizing the hard-case loss L_{hard}^{diff} on the cosine distance map between $\mathbf{F}_s^{diff}$ and $\mathbf{F}_t$.

5. Experimental results

In this study, all experiments were conducted using an NVIDIA GeForce RTX 4060 Ti equipped with 16 GB of memory. The code was implemented in PyTorch version 2.0.1, and Python 3.10.11 was used. All our models were trained under identical hyperparameters: we utilized the Adam optimizer with a learning rate of 1e-4, set the number of training epochs to 160, and employed a batch size of one.

In our research, we assessed the performance of anomaly detection at both the image and pixel levels, with each evaluation providing distinct insights into the model's capabilities. Pixel-level evaluation focuses on the model's precision in identifying specific defect and contamination areas within an image. In contrast, image-level evaluation measures the model's overall ability to detect the presence of defects and contamination in images. For example, a model may excel in image-level detection by accurately identifying images containing anomalies, but it may fall short in pixel-level precision by missing minor defects, thus reducing its pixel-level performance. Conversely, a model with high pixel-level accuracy might correctly identify all anomalies, including the smallest ones, but it may struggle with image-level performance as it may frequently label very small areas as defects.

In our evaluation, we computed two common metrics for anomaly detection: the Area Under the Receiver Operating Characteristic curve (AUROC) and the F1 score. The AUROC offers a comprehensive measure of the model's capability to differentiate between normal and anomalous samples at various threshold settings, while the F1 score provides an assessment of the model's precision and recall. We also evaluated the computational efficiency of the proposed method. Specifically, the multiply-and-accumulate operations (MACs) total 66.196G for an input image size of (384, 384), and it achieves approximately 56 FPS on an NVIDIA GeForce RTX 4060 Ti with 16 GB.

We present the performance of each data augmentation strategy, corruption model, and transfer knowledge strategy individually in Table 2. In the vanilla training strategy, the network is trained using the mean value of D_{cosine} as the loss function, whereas all other strategies employ the hard feature loss. The experimental results indicate that using the hard feature loss consistently enhances the performance of the anomaly detection network across different cup types. This improvement is expected, especially since cups often possess numerous textureless areas, which are easier for the student network to imitate based on the teacher network's output. However, this can cause the student network to become stuck at local optima. The hard feature loss helps direct the network's attention toward details that are not yet well-reconstructed, thereby enhancing the anomaly detection performance. The experimental results also indicate that the knowledge transfer approaches can significantly enhance the model's performance compared to the models trained with data augmentation and corruption models.

The effectiveness of knowledge transfer approaches, when integrated with various data augmentation strategies and corruption models, is detailed in Table 3. Despite their potential, transfer noise techniques fall short in terms of performance when used in conjunction with other data augmentation strategies and corruption models, displaying inconsistent results. In contrast, models trained with samples generated by diffusion models demonstrate significantly improved performance when paired with other training strategies. Nevertheless, the most notable improvement is observed when models are trained exclusively with a combination of data generated by diffusion models with brightness-contrast adjustments and random rotation for data augmentation. It appears that introducing latent noise and stained-shape noise does not uniformly enhance the model's performance in these instances. This observation is logical, considering that samples generated by diffusion networks tend to be more realistic and closely mimic actual defects and contamination.

We present the quantitative results of the baseline methods and the best combination scheme (brightness-contrast adjustment, transfer noise, and stained-shape noise) in Figs. 8–11. These figures include the segmentation results and the corresponding heatmaps (which show the probability of anomalies). In this work, we apply the **cv2.applyColorMap** function from the OpenCV library to visualize the anomaly map using the **COLORMAP_JET** scheme. The **COLORMAP_JET** scheme is a color map choice that starts with blue at the lowest end of the scale and transitions through shades of blue, green, yellow, and orange, ending with red at the highest end. This means cooler colors (blues) indicate a lower likelihood of anomalies, while warmer colors (reds) indicate a higher likelihood of anomalies.

These results demonstrate that EfficientAD is not suitable for training with a small number of samples, even when employing the hard-case loss function. However, it's noteworthy that the hard-case loss function significantly improves the training of the anomaly detection network in the proposed network. On the other hand, cosine distance shows better performance when trained with fewer samples, as evidenced in the reverse distillation approach and our network. However, the heatmap produced by the reverse distillation does not distinctly differentiate between normal details and anomalies. In contrast, our proposed method, which includes data augmentation and a data corruption model, results in much clearer heatmap visualizations, making it easier to identify anomalies. Furthermore, the proposed method, with data augmentation and the transfer of knowledge about defects in other types of cups, can also work effectively even with just a single sample.

Even though the proposed approach can enhance the performance of the anomaly detection model on a training set with just a few images compared to the state-of-the-art methods, it is still challenging to pinpoint the discoloration areas on the reusable cups since the differences in the color images are not significant. Hyperspectral cameras could be used to provide more spectral bands, which might be beneficial for detecting discolorations. Liu et al. Liu et al. (2022) have demonstrated the effectiveness of combining hyperspectral imaging with advanced data analysis techniques, such as autoencoders and self-supervised classifiers, for anomaly detection in food items like strawberries. Similarly, hyperspectral imaging has proven useful for the early detection of mold in food products, as shown by Farrugia et al. Farrugia et al. (2021). As Bleszynski et al. Bleszynski et al. (2020) indicated, hyperspectral imaging can be utilized to visualize polymer damage that cannot be observed in normal color images. Medus et al. Medus et al. (2021) employed convolutional neural networks (CNNs) on hyperspectral images to detect contaminants and defects in food packaging. These studies show that an increased number of spectral bands can enhance detection capabilities for subtle variations in color and texture that are otherwise difficult to detect in normal color images.

6. Conclusion

The hard feature loss guides the student network to focus on details that are poorly reconstructed by the teacher network, thereby enhancing its ability to detect anomalies. This strategy represents a significant improvement over the conventional training method, which used the average of L_{hard} as the loss function.

Moreover, introducing random variations in brightness, contrast, and rotation has significantly increased the accuracy of anomaly detection. This technique introduces a level of variability that trains the network to recognize anomalies under a broader range of conditions, thereby enhancing its robustness and reliability. Furthermore, the strategic transfer of known defects and contaminations based on a diffusion model has proven effective in improving the performance of the detection network. This method leverages existing knowledge of defects from current types of cups to better prepare the network for a new type of cup, as demonstrated by our experimental results. Furthermore, the proposed anomaly detection framework is not limited to reusable foodware; it can also be extended to other problems such as detecting mold or other quality defects in fruits, or identifying defects in various mass-produced products.

Nonetheless, challenges remain in detecting subtle discolorations or minor defects, which will need to be addressed in future work. This could involve enhancing the architecture of the anomaly detection network or utilizing advanced sensors, such as hyperspectral cameras, to capture these issues more clearly. Furthermore, as we collect more data, we can also employ the diffusion network to generate additional data for the supervised learning process and combine it with unsupervised training to achieve more accurate classification. Thus, the anomaly detection framework can gradually improve both its accuracy and robustness.

CRediT authorship contribution statement

Anh Minh Truong: Visualization, Data curation, Writing – original draft, Writing – review & editing, Investigation, Formal analysis, Software, Methodology, Writing – review & editing. **Hiep Quang Luong:** Conceptualization, Writing – review & editing, Resources, Methodology, Validation, Supervision, Project administration, Funding acquisition.

Declaration of AI and AI-assisted technologies in the writing process

During the preparation of this work the author(s) used ChatGPT in order to check the grammatical mistakes. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

TRUONG MINH ANH reports financial support was provided by Europees Fonds voor Regionale Ontwikkeling. HIEP LUONG reports financial support was provided by Europees Fonds voor Regionale Ontwikkeling. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

This research has been carried out within the EFRO Vision2Reuse project, with the support from EFRO (Europees Fonds voor Regionale Ontwikkeling), under grant agreement No. 1508. The samples have been collected at AMAB, a Belgian social enterprise for customised work.

References

Batzner, K., Heckler, L., König, R., 2023. Efficientad: accurate visual anomaly detection at millisecond-level latencies. arXiv:2303.14535.

Bergmann, P., Fauser, M., Sattlegger, D., Steger, C., 2020. Uninformed students: student-teacher anomaly detection with discriminative latent embeddings. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4182–4191.

Bergmann, P., Lowe, S., Fauser, M., Sattlegger, D., Steger, C., 2019. Improving unsupervised defect segmentation by applying structural similarity to autoencoders. In: Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications.

Bleszynski, M., Mann, S., Kumosa, M., 2020. Visualizing polymer damage using hyperspectral imaging. Polymers 12.

Cohen, N., Hoshen, Y., 2020. Sub-image Anomaly Detection with Deep Pyramid Correspondences arXiv preprint arXiv:2005.02357.

Collin, A.S., De Vleeschouwer, C., 2021. Improved anomaly detection by training an autoencoder with skip connections on images corrupted with stain-shaped noise. In: 2020 25th International Conference on Pattern Recognition (ICPR), pp. 7915–7922.

Deng, H., Li, X., 2022. Anomaly detection via reverse distillation from one-class embedding. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 9727–9736.

Ehret, T., Davy, A., Morel, J.M., Delbracio, M., 2019. Image anomalies: a review and synthesis of detection methods. J. Math. Imag. Vis. 61, 710–743.

Farrugia, J., Griffin, S., Valdramidis, V.P., Camilleri, K., Falzon, O., 2021. Principal component analysis of hyperspectral data for early detection of mould in cheeselets. Curr. Res. Food Sci. 4, 18–27.

Fernando, T., Gammulle, H., Denman, S., Sridharan, S., Fookes, C., 2021. Deep learning for medical anomaly detection – a survey. ACM Comput. Surv. 54.

Glodek, M., Schels, M., Schwenker, F., 2013. Ensemble Gaussian mixture models for probability density estimation. Comput. Stat. 28, 127–138.

Gudovskiy, D.A., Ishizaka, S., Kozuka, K., 2021. Cflow-ad: real-time unsupervised anomaly detection with localization via conditional normalizing flows. In: 2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), pp. 1819–1828.

Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W., 2022. LoRA: low-rank adaptation of large language models. In: International Conference on Learning Representations. URL: https://openreview.net/forum?id=nZeVKeeFYf9.

Hu, T., Zhang, J., Yi, R., Du, Y., Chen, X., Liu, L., Wang, Y., Wang, C., 2023. Anomalydiffusion: Few-Shot Anomaly Image Generation with Diffusion Model. ArXiv abs/2312.05767.

Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., Aila, T., 2020. Analyzing and Improving the Image Quality of StyleGAN. Proc. CVPR.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2017. Imagenet Classification with Deep Convolutional Neural Networks. Association for Computing Machinery, New York, NY, USA, pp. 84–90. https://doi.org/10.1145/3065386.

Latecki, L.J., Lazarevic, A., Pokrajac, D., 2007. Outlier detection with kernel density functions. In: Perner, P. (Ed.), Machine Learning and Data Mining in Pattern Recognition. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 61–75.

Lin, W., Tong, T., Gao, Q., Guo, D., Du, X., Yang, Y., Guo, G., Xiao, M., Du, M., Qu, X., Initiative, A.D.N., 2018. Convolutional neural networks-based mri image analysis for the alzheimer's disease prediction from mild cognitive impairment. Front. Neurosci. 12, 777.

Liu, Y., Zhou, S., Wu, H., Han, W., Li, C., Chen, H., 2022. Joint optimization of autoencoder and self-supervised classifier: anomaly detection of strawberries using hyperspectral imaging. Comput. Electron. Agric. 198, 107007.

Liu, Z., Zhou, Y., Xu, Y., Wang, Z., 2023. Simplenet: a simple network for image anomaly detection and localization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 20402–20411.

Medus, L.D., Saban, M., Francés-Víllora, J.V., Bataller-Mompeán, M., Rosado-Muñoz, A., 2021. Hyperspectral image classification using cnn: application to industrial food packaging. Food Control 125, 107962.

Mei, S., Wang, Y., Wen, G., 2018. Automatic fabric defect detection with a multi-scale convolutional denoising autoencoder network model. Sensors 18.

Nalisnick, E.T., Matsukawa, A., Teh, Y.W., Görür, D., Lakshminarayanan, B., 2019. Do deep generative models know what they don't know?. In: 7th International Conference on Learning Representations. ICLR 2019, New Orleans, LA, USA. May 6-9, 2019.

Roth, K., Pemula, L., Zepeda, J., Scholkopf, B., Brox, T., Gehler, P., 2022. Towards total recall in industrial anomaly detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14318–14328.

Saleh, B., Farhadi, A., Elgammal, A., 2013. Object-centric anomaly detection by attribute-based reasoning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).

Schlegl, T., Seeböck, P., Waldstein, S.M., Langs, G., Schmidt-Erfurth, U., 2019. f-anogan: fast unsupervised anomaly detection with generative adversarial networks. Med. Image Anal. 54, 30–44.

Seeböck, P., Orlando, J.I., Schlegl, T., Waldstein, S.M., Bogunović, H., Klimscha, S., Langs, G., Schmidt-Erfurth, U., 2020. Exploiting epistemic uncertainty of anatomy segmentation for anomaly detection in retinal oct. IEEE Trans. Med. Imag. 39, 87–98.

Steger, C., Ulrich, M., Wiedemann, C., 2008. Machine Vision Algorithms and Applications.

Touvron, H., Cord, M., Douze, M., Massa, F., Sablayrolles, A., Jegou, H., 2021. Training data-efficient image transformers & distillation through attention. In: International Conference on Machine Learning, pp. 10347–10357.

Wang, C.Y., Bochkovskiy, A., Liao, H.Y.M., 2022. YOLOv7: Trainable Bag-Of-Freebies Sets New State-Of-The-Art for Real-Time Object Detectors arXiv preprint arXiv:2207.02696.

Yang, B., Gu, S., Zhang, B., Zhang, T., Chen, X., Sun, X., Chen, D., Wen, F., 2023. Paint by example: exemplar-based image editing with diffusion models. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 18381–18391. https://doi.org/10.1109/CVPR52729.2023.01763.

Yu, J., Zheng, Y., Wang, X., Li, W., Wu, Y., Zhao, R., Wu, L., 2021. Fastflow: Unsupervised Anomaly Detection and Localization via 2d Normalizing Flows arXiv:2111.07677.

Zhang, G., Cui, K., Hung, T.Y., Lu, S., 2021. Defect-gan: high-fidelity defect synthesis for automated defect inspection. In: 2021 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 2523–2533. https://doi.org/10.1109/WACV48630.2021.00257.