Applied Intelligence
https://doi.org/10.1007/s10489-023-05177-0

Exploiting CNN's visual explanations to drive anomaly detection

Michele Fraccaroli[1] · Alice Bizzarri[1] · Paolo Casellati[1] · Evelina Lamma[1]

Accepted: 12 November 2023

Abstract

Nowadays, deep learning is a key technology for many applications in the industrial area such as anomaly detection. The role of Machine Learning (ML) in this field relies on the ability of training a network to learn to inspect images to determine the presence or not of anomalies. Frequently, in Industry 4.0 w.r.t. the anomaly detection task, the images to be analyzed are not optimal, since they contain edges or areas, that are not of interest which could lead the network astray. Thus, this study aims at identifying a systematic way to train a neural network to make it able to focus only on the area of interest. The study is based on the definition of a loss to be applied in the training phase of the network that, using masks, gives higher weight to the anomalies identified within the area of interest. The idea is to add an *Overlap Coefficient* to the standard cross-entropy. In this way, the more the identified anomaly is outside the *Area of Interest* (AOI) the greater is the loss. We call the resulting loss *Cross-Entropy Overlap Distance* (CEOD). The advantage of adding the masks in the training phase is that the network is forced to learn and recognize defects only in the area circumscribed by the mask. The added benefit is that, during inference, these masks will no longer be needed. Therefore, there is no difference, in terms of execution times, between a standard Convolutional Neural Network (CNN) and a network trained with this loss. In some applications, the masks themselves are determined at run-time through a trained segmentation network, as we have done for instance in the "Machine learning for visual inspection and quality control" project, funded by the MISE Competence Center Bi-REX.

Keywords Visual explanation · Anomaly detection · Visual inspection · Defect localization

1 Introduction

Nowadays, deep learning enables the automation of many industrial tasks, reducing dependence on human intervention and improving efficiency. For example, automation of production lines, data management of industrial sensors, and predictive maintenance are some of the main applications. Industry benefits from deep learning for inspection and quality control. Deep neural networks can detect defects or anomalies in goods more accurately and quickly than manual inspection. Anomaly detection in industrial image data is of utmost importance for many tasks in computer vision [1]. However, training deep learning models requires large amounts of high-quality data, and in the field of surface analysis, very often images acquired in the industrial environment contain some sections that are not part of the surface to be inspected [2].

Just think of images of products running on conveyor rollers or connected to other components not subject to inspection or simply images of the edge of the product which inevitably incorporates part of the background. In many cases, if we know the shape of a product to be inspected, we can simply use some traditional image processing techniques to remove the useless parts from the images. But, in other cases, we don't know the exact shape of our product or where the background appears in the image.

In order to focus on relevant points of an image, this work proposes and tests a new approach to identify a systematic way to train a CNN that focuses only on the area of interest. To do that, we identify the most important pixels in the images for classification according to a CNN. After that, we calculate how much these pixels overlap with the mask that is provided, for each image, during the training phase. The computed

✉ Michele Fraccaroli
michele.fraccaroli@unife.it

Alice Bizzarri
alice.bizzarri@unife.it

Paolo Casellati
paolo.casellati@edu.unife.it

Evelina Lamma
evelina.lamma@unife.it

[1] DE - Department of Engineering, University of Ferrara, via Saragat 1, 44122 Ferrara, Italy

Published online: 12 December 2023

overlap value is added to the loss of the network, to force the network to recognize the most important pixels, only within the area marked by the masks.

The rest of the paper is organized as follows: Section 3 describes the problem and the scenario where this work is placed, Section 4 describes the idea behind this work, the mathematical formulation, and the algorithm for the custom loss developed in this paper. Section 5 illustrates the experiments and the results obtained on the various datasets. In Sections 2 and 6 we present related work and conclusions respectively.

2 Related work

In Machine Learning, anomaly detection has long been an issue of great interest, especially in Industry 4.0 where identifying defects is one of the major tasks of computer vision. Various articles survey anomaly detection in the literature [3, 4]. The aim of our work focuses primarily on the use of CNNs for structural defect detection during the monitoring of manufacturing line.

The vast majority of the CNN-based approaches have been used to study anomalies in the whole image area. Weimer D, et al. in [5] investigate CNNs in order to overcome the difficulties of manually redefining a specific feature representation for each new industrial inspection problem. In [6] the authors show how CNN with Triplet Loss [7] can be used to identify anomalies in the industrial environments.

In the context of anomaly detection in industrial images, Samet Akcay et al. [8] introduced an approach based on Generative Adversarial Network (GAN) [9]. This approach was designed to address a common challenge in the industry, namely when the sample of positive examples (usually representing anomalies) is limited, while negative examples (normal images) are in large numbers. The GAN learns the distribution of the class of interest and uses the difference between a reconstructed image and an input image to detect anomalies. This methodology has proven effective for anomaly detection, even in scenarios where the number of positive examples is low. An and Cho in their study [10] use a Variational Autoencoder (VAE) [11] for anomaly detection. However, it is important to note that GAN or Variational Autoencoders (VAE)-based approaches tend to be more effective in reconstructing simple anomalies. They may encounter difficulties when dealing with images that contain noise, such as background, commonly found in industrial environments. This is a point of challenge that needs further consideration.

Ferrari et al. [12] illustrate an architecture consisting of a GAN to perform the reconstruction and denoising processes and a model for image segmentation capable of detecting defects. The discriminative network is trained using an AOI for each image in the training dataset. The network learns in which area the defects are relevant. In this way, the use of pre-processing algorithms is reduced. Finally, the model was tested on MVTec's anomaly detection dataset and a large industrial dataset.

In [13] Yong Moon et al. show the importance of using Class Activation Maps (CAMs) to check if the neural network focuses on the area of interest. The authors analyze the CNN architecture in detail using CAM images along with several evaluation metrics to optimize the CNN. Recently, path imaging has been shown to be effective in the segmentation and recognition of anomalies [14, 15]. In [16], the authors introduce the use of Grad-CAM to construct a self-supervised method to remove image noise for robust anomaly detection. Venkataramanan et al. [17] use the activation map to guide autoencoder training by reducing network attention in abnormal areas and increasing attention in normal areas in order to strengthen anomaly detection. Song et al. [18] proposed an interesting methodology based on an Anomaly Segmentation Network (AnoSeg). This network was developed to generate an anomaly map, thus allowing anomalous regions in the image to be effectively segmented. The AnoSeg approach represents a significant contribution as it addresses the challenge of not only detecting anomalies but also segmenting them precisely. This is particularly useful in industrial contexts where it is important to identify not only the presence of anomalies but also their spatial extent. However, it should be noted that AnoSeg, like other neural network-based methods, can also be affected by the presence of noise or background in the image, which can pose a significant challenge in the industrial environment.

In our approach, we use Grad-CAM [19] to add a penalty when the neural network detects an anomaly outside the AOI. Our methodology differs from the previously mentioned methods because the neural network focuses on distinguishing imperfections in a specific area of the image and not on the entire image. This allows the CNN to learn to distinguish anomalies in the area of interest from noise generated by a heterogeneous background, thus addressing some of the challenges associated with anomaly detection in industrial images.

These detection systems can also be applied in contexts other than anomaly detection, e.g. in the field of marine detection, several algorithms have been developed that can detect objects by removing noise, through attention-based spatial pyramid pooling networks and bidirectional feature fusion strategy [20, 21]. In [22], the authors use a Multi-Path DCNN Model, dividing the image into three areas of interest by carefully examining each part.

3 Problem description

As mentioned before, the problem is due to the heterogeneity that can occur in the images to be analyzed in an industrial environment. In some cases, it is not possible to perfectly isolate the piece of the surface to be analyzed due to the shape of the object or the environment in which the image of the object is acquired. These problems can bring a lot of useless information into the dataset which should still be processed by vision systems (neural networks in this case). Figure 1, on the left, shows a representation of the surface with intrinsic features like a tapped hole for a screw, which could be recognized as a defect since this feature is not present in all images and not always in the same location. On the right, there is a representation of a curved surface that, due to the curvature itself, can present dark areas that could have light refractions and might lead a neural network to mistake them for defects [23]. This useless information could alter the result of the network making it inefficient or unusable. Figure 2 shows an example that well describes the problem.

To solve this problem, we need to focalize the vision system on a specific *Area of Interest* (AOI) making sure to weigh more the contribution of the information contained in this area than in the rest of the image.

This problem is like the task of instance/image segmentation but, unfortunately, in the industrial environment and the anomaly detection field, we don't know a priori the defect. Thus, we can't generate the masks that highlight the important parts of the images and use these as labels to train a segmentation network. For this reason, we focus on standard Convolutional Neural Networks (CNNs) for performing a binary classification to classify the images with anomalies.

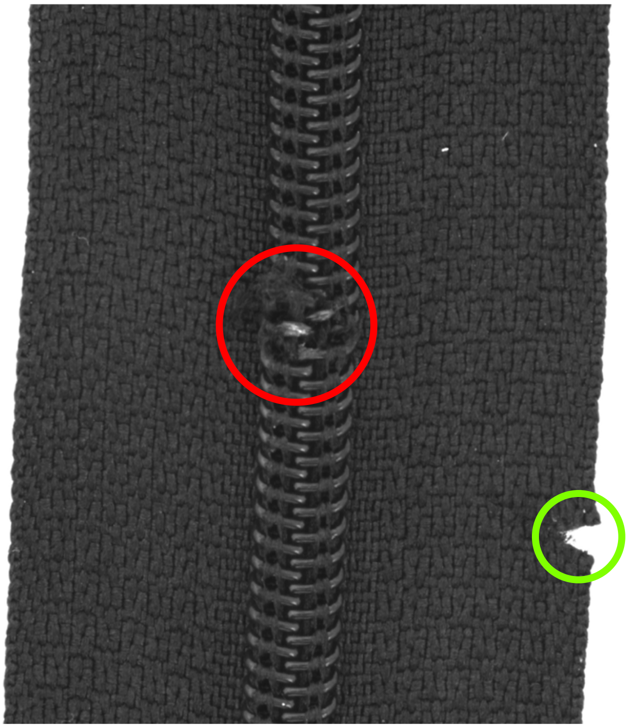


Fig. 2 Example of the image taken from the MVTec AD dataset [24, 25] with two defects: one inside (red circle) and one outside (green circle) of the area of interest respectively. The damage outside of the area of interest, for classification purposes, must be considered not a defect

4 Cross-entropy overlap distance

The idea is to obtain a value that expresses how much two areas within an image overlap. As an overlap value, we use the

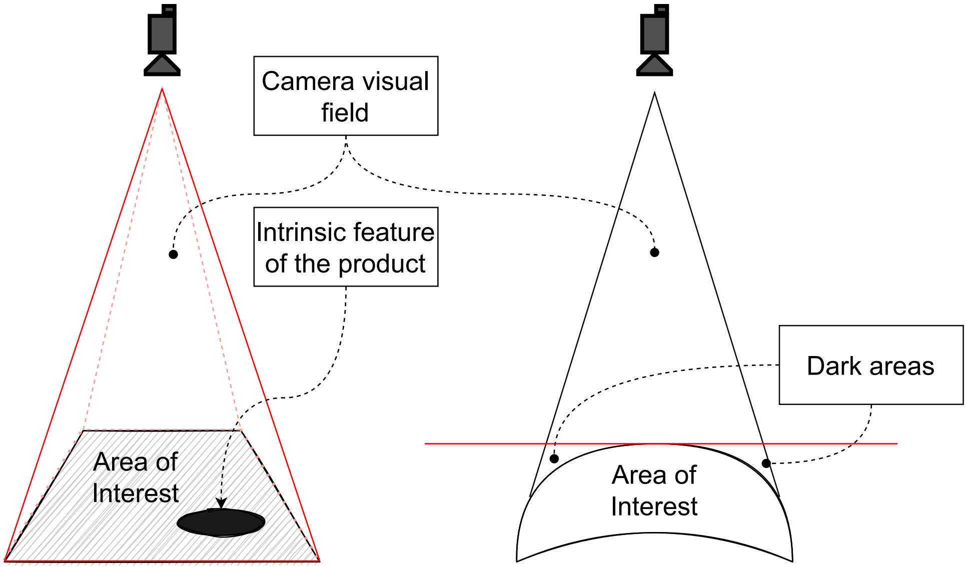


Fig. 1 Visual examples of possible problems encountered during surface analysis

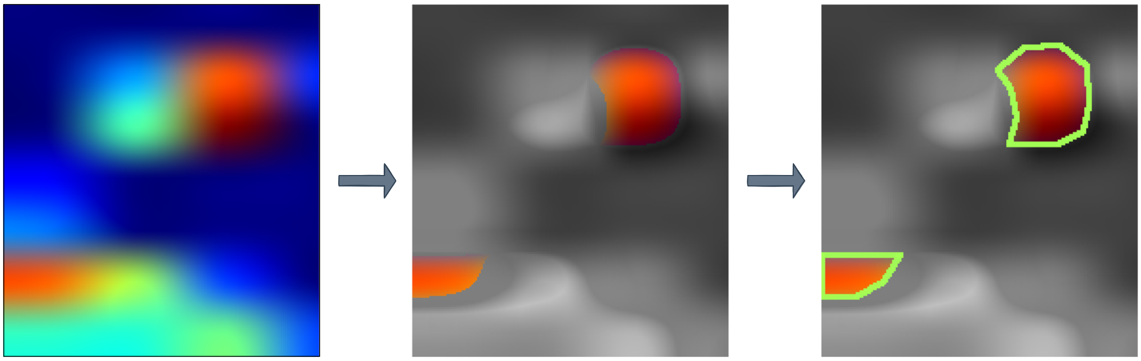


Fig. 3 Extraction of the hottest pixels

Overlap coefficient (also known as the Szymkiewicz - Simpson coefficient) [26]. The objects for which we are going to calculate the overlap will be the mask provided during training (present in the dataset) and the region of the image that CNN believes is most significant for the recognition of that image. To do this, we exploit an *explanation algorithm* called *Gradient-weighted Class Activation Mapping* (Grad-CAM) [19]. It allows us to identify which area of the image is most involved in the network decision. Then, at the end of each forward pass of the network's training phase, we calculate an *heatmap* that highlights the most important pixels in the input image (the hottest pixels) with *Grad-CAM*. After that, we extract the hottest pixels (see Fig. 3) and we calculate how much these hottest pixels are overlapped with the mask. The greater the overlap, the lower the penalty applied to the loss. This will allow the network to learn which area of the image to focus on, so masks will no longer be needed in the inference phase. Figure 4 shows the training phase with CEOD.

4.1 Visual explanation by Grad-CAM

Grad-CAM [19] is a localization technique based on the *Class Activation Mapping* (CAM) algorithm [27] that generates visual explanations for any CNN without requiring changes or re-training. In order to generate a class-discriminative heatmap, Grad-CAM computes the gradient of the output score for class cls, the output (out_{cls}) calculated before the last (softmax) activation w.r.t. the feature map activations of the last convolutional layer. The global average pooling of these gradients is calculated to obtain the neuron importance weights α_{cls}:

$$\alpha_{cls} = \frac{1}{P} \sum_{i} \sum_{j} \frac{\partial out_{cls}}{\partial A} \tag{1}$$

where the $\sum_i \sum_j$ represent the global-average pooling and P represents the number of pixels in the feature map. Finally,

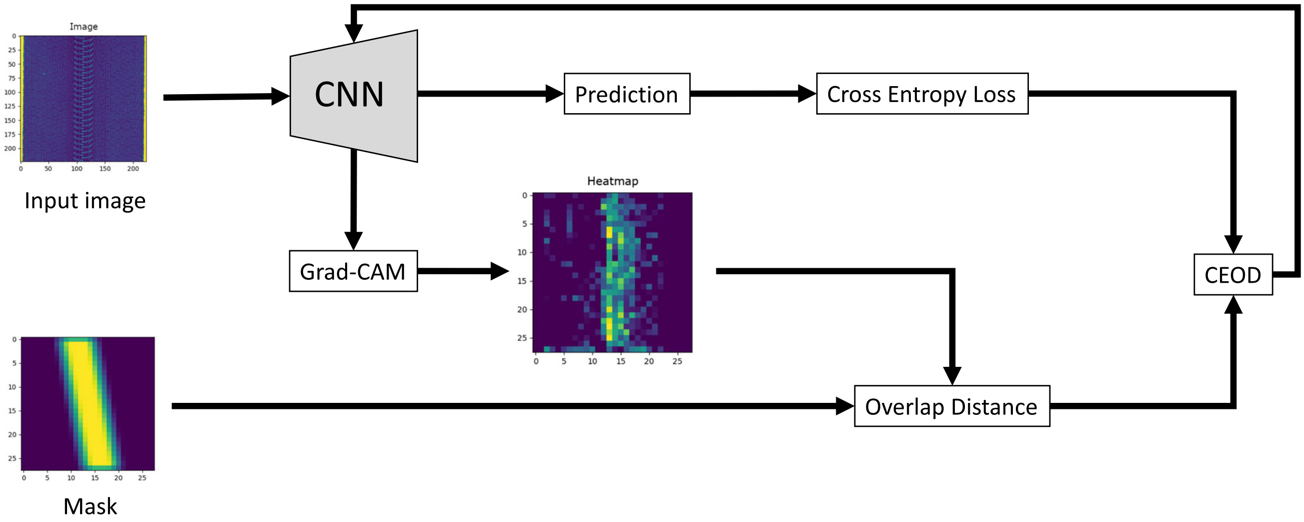


Fig. 4 Cross-Entropy Overlap Distance training phase

a weighted combination of activation maps is performed, followed by a ReLU to obtain the heatmap:

$$L_{cls}^{HeatMap} = ReLU\left(\sum_{k} \alpha_{cls} A\right) \tag{2}$$

For more details, see the work of R.R. Selvaraju et al. [19].

4.2 Mathematical formulation

Overlap Coefficient equation is:

$$overlap_c(A_d, A_{gt}) = \frac{|A_d \cap A_{gt}|}{min(|A_d|, |A_{gt}|)} \tag{3}$$

where A_d and A_{gt} are the areas obtained through *Grad-CAM* and the segmentation mask (or area of the ground truth) respectively. A_{gt} is obtained with a manual segmentation or using a previously trained *segmentation neural network* [28–30]. Figure 5 shows a graphical representation of A_d and A_{gt}. In this case, if A_d is a subset of A_{gt} or the converse, the *Overlap Coefficient* is 1. If we want to add this term to the loss function of the neural network, we need to negate the Overlap Coefficient. Applying the negation of the logarithm we obtain a new value that we have called *Overlap Distance* (OD), expressed by (4):

$$OD(A_d, A_{gt}) = -\ln\left(\frac{|A_d \cap A_{gt}|}{min(|A_d|, |A_{gt}|)}\right) \tag{4}$$

In this way, when A_d is a subset of A_{gt}, we obtain the $-\ln(1)$, and OD becomes 0, giving no contribution to the loss. The logarithm was introduced because it offers less penalty for small differences between predicted and corrected values. When the difference is large, the penalty will be higher. To optimize our CNN also w.r.t. this further aspect, we need to add this new term to the *Cross-Entropy* loss [31], as described by the following equation:

$$ce = -\frac{1}{N}\sum_{i=1}^{N} y_i \log(p(y_i)) \tag{5}$$

where N is the number of examples, y_i and $p(y_i)$ are the label and the output of the network for the *i-th* example respec-

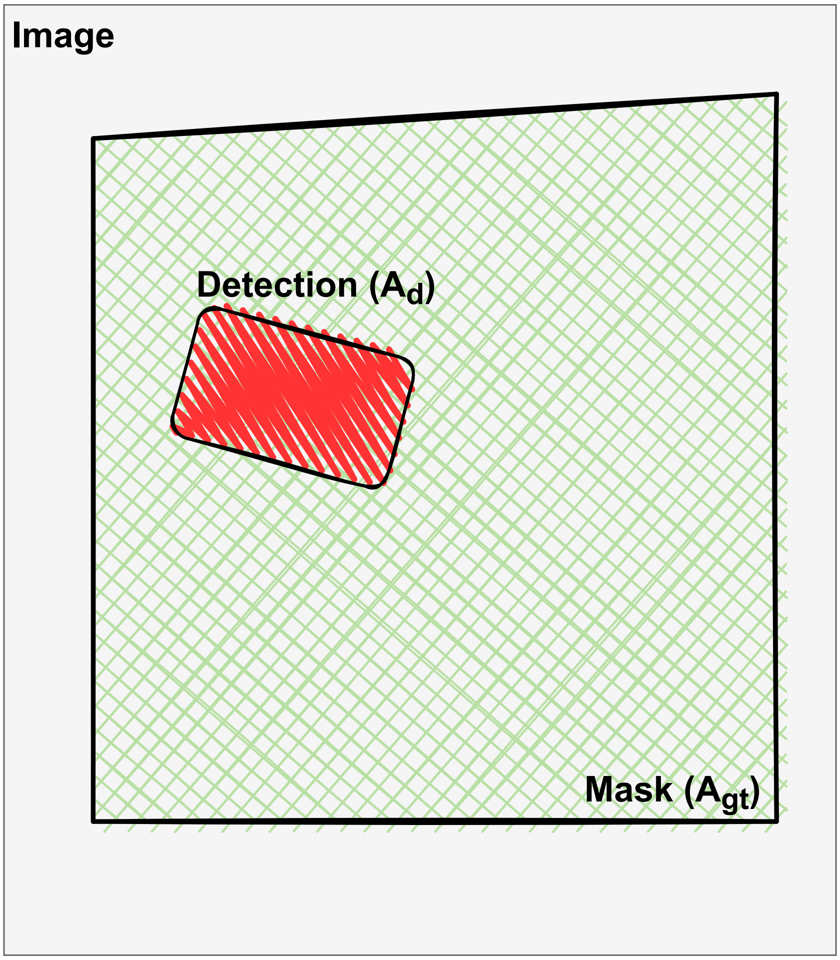


Fig. 5 Stylized sample image with A_d and A_{gt} as area of the network detection and mask respectively

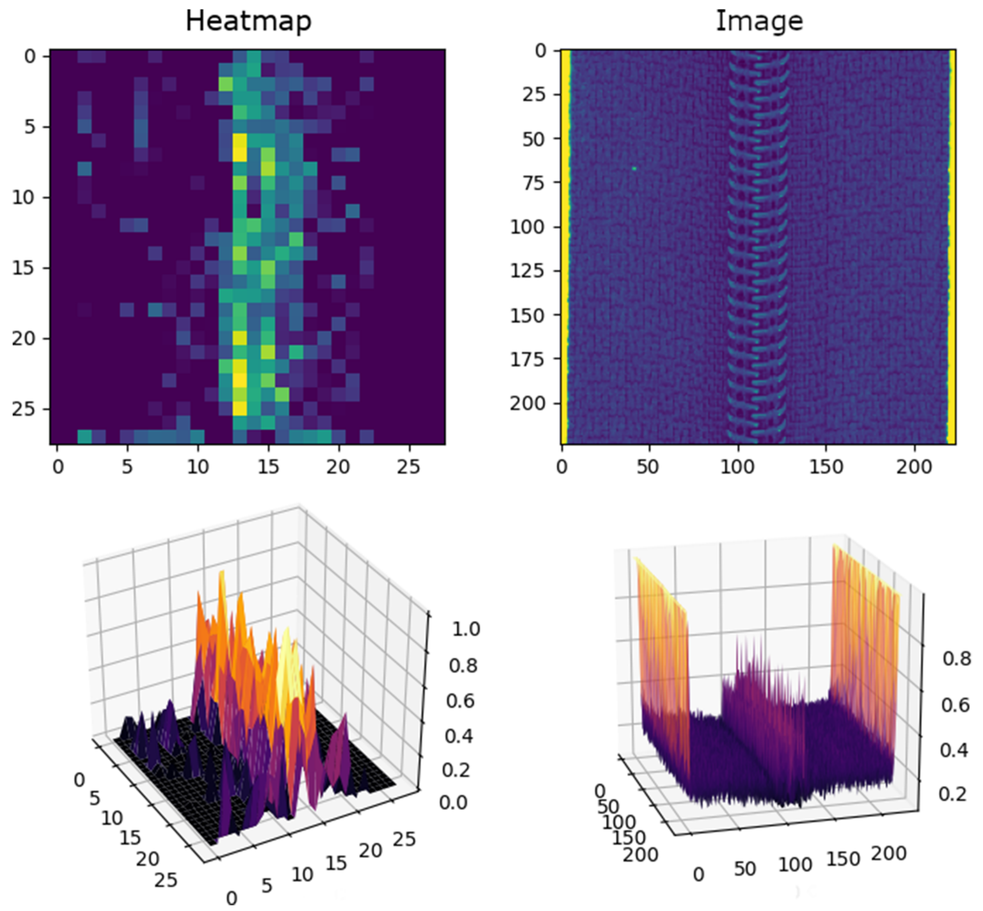


Fig. 6 2D and 3D heatmap (top left and bottom left) obtained with Grad-CAM from an image (top right and bottom right)

tively. This is necessary to take into account the contribution of the classification task to the overall loss. We thus obtain the *Cross-Entropy Overlap Distance* (CEOD) that is:

$$CEOD = ce + OD(A_d, A_{gt}) \tag{6}$$

$$CEOD = -\frac{1}{N}\sum_{i=1}^{N} y_i \log(p(y_i)) + \omega y_i \left(-\ln\left(\frac{|A_d^i \cap A_{gt}^i|}{min(|A_d^i|, |A_{gt}^i|)}\right)\right) \tag{7}$$

The term ω in (7) is a new hyper-parameter to be set which represents the degree of impact of the new term on the overall loss. This term depends on the order of magnitude and on the difference between the two parts of the loss. In our experiments, after different tests, we have set ω to 0.001. The OD part of the loss is also multiplied by y_i to take into account the label of the images. This is because, in the anomaly detection task, the defect-free images (*good* images) do not have specific areas with the hottest pixels but their heat map is rather uniform and with low-intensity levels.

Figure 6 (bottom left image), by filtering the heatmap for extracting the hottest pixels, we can obtain the object (or the area) (then the A_d term) used in (7).

4.3 Algorithm

For exploiting the OD in the training of a CNN, we need to create a custom training loop for obtaining the feature extracted in the last convolutional layer, generating the heatmap, and then using this in CEOD. For obtaining both the classification output and the features extracted from the last convolutional layer, the output of the CNN was modified. Algorithm 1 shows the process behind the custom training loop and the CEOD loss.

As can be seen in line 3 of Algorithm 1, we have applied a convolution with filter $\left(\begin{smallmatrix}.5 & .5 & .5\\ .5 & .5 & .5\\ .5 & .5 & .5\end{smallmatrix}\right)$ to incorporate a *distance* concept in the OD computation. The transformation of the mask after convolution is visible in Fig. 7. Note how, after convolution, there are no more clear differences in height (between the values 1 and 0, see the bottom right and bottom left 3D representations in Fig. 7) but the AOI of the mask becomes more gradual, widening the AOI and allowing us to imple-

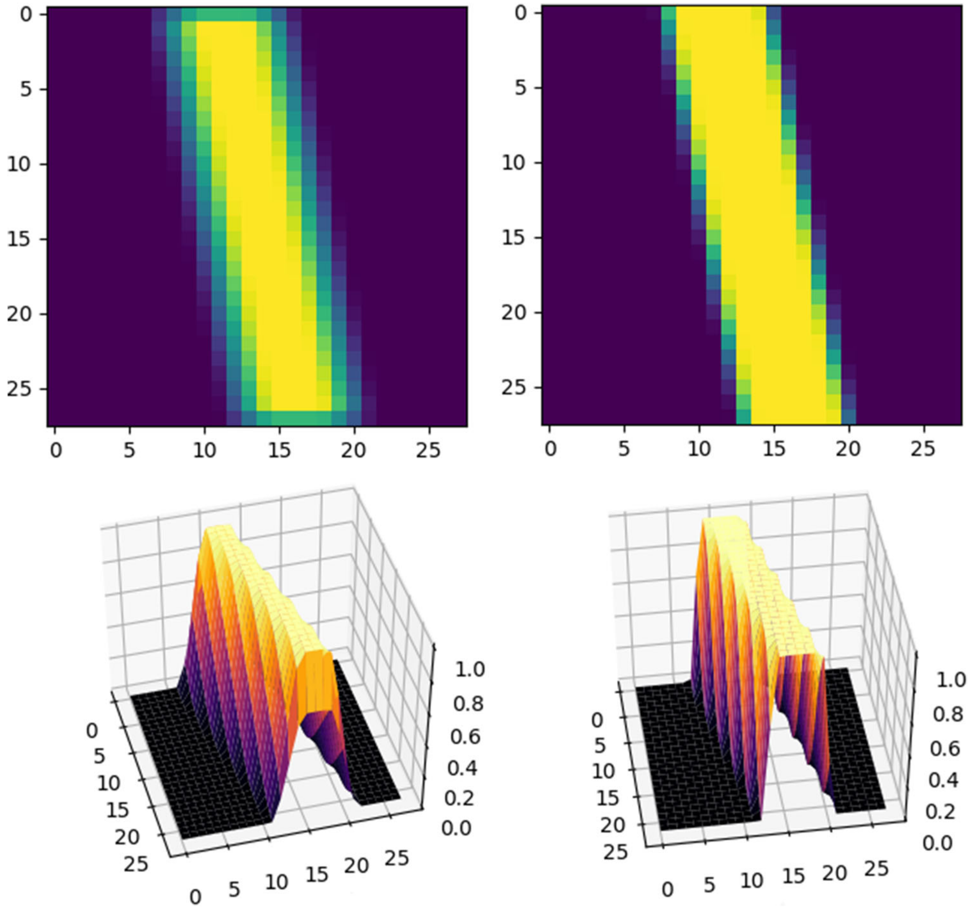


Fig. 7 2D and 3D representation of the filtered mask (top left and bottom left) and the mask inside the dataset (top right and bottom right)

ment the distance computation so that it can be differentiated as the rest of the loss.

The *distance* is important to help the network understand when the detection is far or near the AOI. Following the example of Fig. 8, focusing on the original mask (left side of Fig. 8) we can see that the detection marked with A and B have two different distances (Da and Db) and the distance of B (Db) from the AOI is larger than Da. If we use the original mask in the CEOD, these two detections give the same results because both A and B are multiplied by the same mask value which is zero. Instead, if we exploit the filtered masks, we can see that $A1$ is partially over the AOI, so its contribution to the calculation of the CEOD will be greater than $B1$. The closer the detection is to the AOI, the smaller the distance. Clearly, in case of overlap, the distance will be zero. This contribution leads the network to understand that the further the detection is from the AOI, the worse it is.

Algorithm 1 CEOD loss calculation and custom training loop.

```
Input: $x, y, mask$
$output, feature\_tensor \leftarrow CNN(x, y)$
$heatmap \leftarrow GradCAM(output, feature\_tensor)$
$mask_{filtered} \leftarrow mask * kernel(\begin{smallmatrix} .5 & .5 & .5 \\ .5 & .5 & .5 \\ .5 & .5 & .5 \end{smallmatrix})$
$OD \leftarrow OD(mask_{filtered}, heatmap)$
$loss \leftarrow crossentropy(output, y) + OD$
Perform backpropagation
```

To make the OD part of the loss differentiable, the formulation of the OD became as follows:

$$OD = -\log\left[clip_{\epsilon}\left(\frac{\sum A_d A^*_{gt}}{min(\sum A_d, \sum A^*_{gt})}\right)\right] \quad (8)$$

where $\sum$ is calculated over the pixels of the images, A^*_{gt} represent the convoluted mask, and $clip_{\epsilon}$ is a function that clips the value of the Overlap Coefficient in the interval $[\epsilon, 1-\epsilon]$ to avoid the logarithm returning unacceptable values.

5 Experiments

In this section, we describe the datasets, the general setup, and the results achieved with the experiments. Each experiment was performed on the Marconi100[1] cluster provided

[1] Marconi100: https://www.hpc.cineca.it/hardware/marconi100

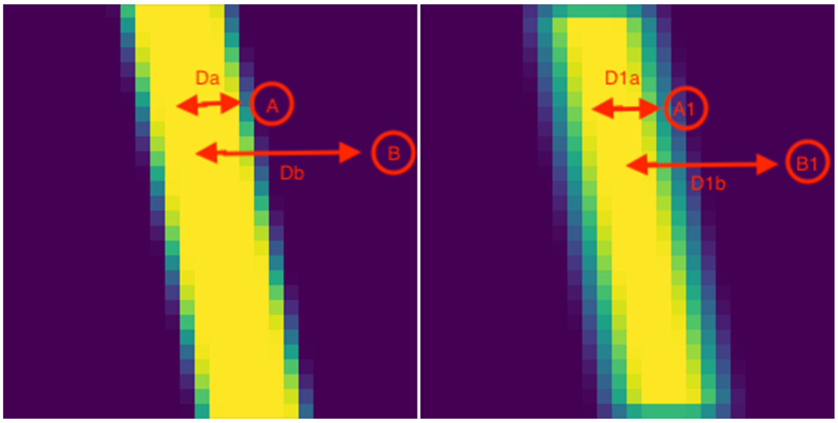


Fig. 8 Original mask (left) and filtered mask (right) with two different detections: A, B, $A1$ and $B1$ respectively. Da, Db, $D1a$, and $D1b$ represent the distance between the detection and the AOI for the original and filtered mask respectively

by Cineca [2], in which each node is equipped with 2 IBM POWER9 AC922 CPUs and 4 NVIDIA Volta V100 with 16GB of RAM and connected to other nodes with NVlink 2.0. We have experimented with two different CNNs: EfficientNet-B0 [32], pre-trained on ImageNet [33], and a custom CNN composed of 8 convolutional blocks (a convolutional block is composed of a convolutional layer, followed by a batch normalization layer) with a max-pooling layer every two blocks. The custom Net has 294994 trainable parameters, 1.33 GFLOPS, and 10,8 ms to an image in inference. The custom CNN was trained from scratch. On the MVTec AD dataset, we have trained both networks with and without the CEOD contribution. Then, we compared the new CEOD loss with the standard classification loss in terms of confusion matrix, accuracy, ROC AUC, and loss.

5.1 Dataset

The experiments were performed on two different datasets. The first dataset is a sub-dataset of MVTec AD [24, 25], specifically, only the zipper images. This sub-dataset was augmented to change its shape and proportions. Figure 9 shows some sample images of this dataset. Each image has an associated binary mask. This dataset is composed of 216 *images without defects* and 184 *images with defects*. Then, in total, there are 400 images in the dataset. The training proportion is 70/20/10 for training, validation, and testing. This dataset was chosen because it is representative of the problem in question. The images in this dataset have flaws found on both the zipper and the fabric on the outside of the zipper. To comply with the problem, we consider only the images that have defects on the zip. Then, all images with no flaws on the zipper but surrounding fabric were relabelled as non-defective.

The second dataset used in these experiments is provided by an Italian company. Unfortunately, it is not possible to give details about this dataset due to an NDA signed with the company in the project. But this latter dataset is a real industrial dataset. This dataset is composed of 2818 *images without defects* and 2246 *images with defects*. In total, there are 5064 images in the dataset. The training configuration is always 80/20 for training and validation. The test set has 893 images. 467 *images without defects* and 426 *images with defects*. The images represent the surface of a product developed by this company. This product has a round shape and a clear reflective smooth surface. For this reason, a light pattern has been applied to these products using a special illuminator to bring out the defects that occur on the surface of the product. The application of this light pattern, due to the reflective surface, causes random scattering effects that are visually comparable to defects. These effects are rarely the same.

In the field of anomaly detection, due to data imbalance, it is customary to augment data to reduce the gap between classes. However, this approach can lead to several problems, such as overfitting, and involves increasingly sophisticated augmentation strategies; this is a hot topic in the literature [34–36]. Therefore we chose not to implement data augmentation, and as described in the next section, state-of-the-art results were obtained

5.2 Results

For each experiment, as mentioned before, EfficientNet-B0 was pre-trained on ImageNet. For our experiments, we have tested both transfer learning and fine-tuning. The experiments show that fine-tuning gives significantly better results than transfer learning. This is due to the fact that the network was trained on ImageNet with a standard loss (categorical cross-entropy). Therefore, re-training only the last dense layer may not be sufficient to be able to enhance the contribution of the new loss. For our experiments, we have performed fine-tuning by unfreezing the last 20 convolutional layers. Various other network configurations will be explored in our future work to assess the impact of architecture on perfor-

[2] Cineca website: https://www.cineca.it/

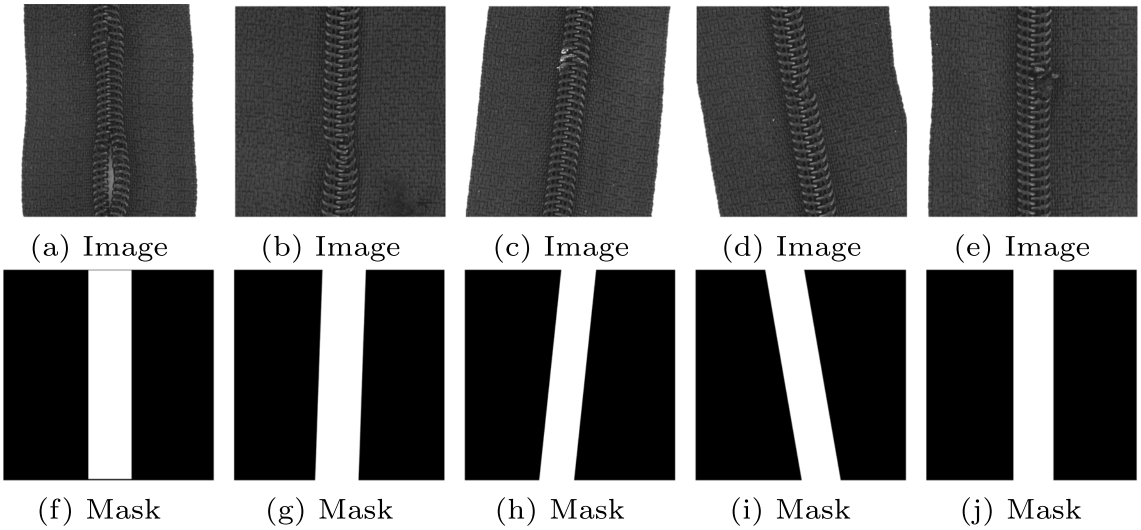


Fig. 9 Example of the augmented zip dataset

mance. The custom CNN, instead, was trained from scratch by adopting the new loss proposed. All experiments were performed with ω at 0.001, batch size at 32, and adamax with learning rate at 0.002.

5.2.1 MVTec AD Dataset

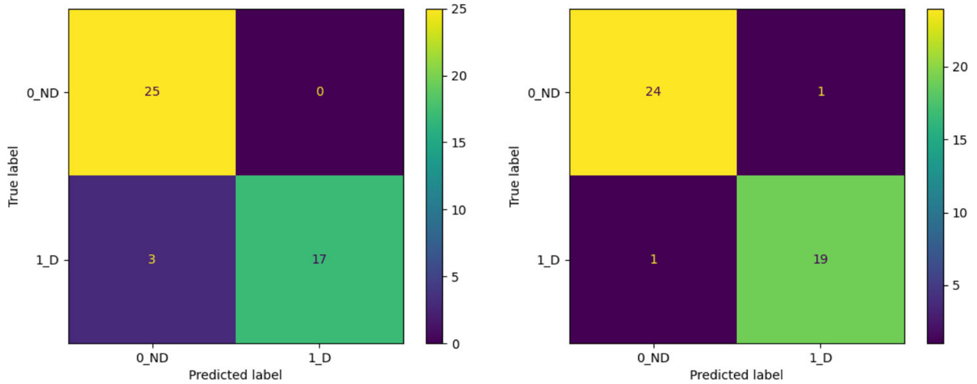
Table 1 shows the results of EfficientNet-B0 and the custom CNN trained on the MVTec AD dataset. Both networks were trained with standard loss (Categorical Cross-Entropy) and with our CEOD. As can be seen, in both cases, the application of the new OD can bring the networks to achieve better results in the validation phases. Figure 10 shows the results of EfficientNet-B0 on the test set. From the confusion matrices, we can see that, by the application of OD on the loss (thus using the CEOD loss), the network obtains better results in terms of defect identification at the expense of a slight worsening in the identification of non-defective ones. The network trained with CEOD reaches 95.5% accuracy and 0.95 AUCROC. Indeed, EfficientNet-B0 with classical cross-entropy reaches 93.3% of accuracy and 0.925 AUCROC. Figure 11 shows the results of custom CNN on the test set. The custom CNN trained with CEOD obtains 73.3% accuracy and 0.74 AUCROC in the same test set. The custom CNN trained with standard cross-entropy reaches 48.8% of accuracy and 0.53 AUCROC. Figure 12 shows an example of a heatmap produced by a network trained classically and the one trained with CEOD. The time spent performing 1K training cycles with EfficientNet-B0 trained with standard cross-entropy is 2h 28m 24s. The time spent performing 1K training cycles with EfficientNet-B0 trained with CEOD is 2h 36m 29s. The time spent with the custom CNN to perform 100 training cycles with standard loss is 4m 30s versus 4m 28s for the custom CNN trained with CEOD.

5.2.2 Industrial dataset

Table 2 shows the results of EfficientNet-B0 and the custom CNN on the industrial real-case dataset. From Table 2, we can see that the network trained with CEOD is better in terms of accuracy in the validation phase. We can also note the improvement of the network trained with fine-tuning w.r.t. the network trained with only transfer learning. The slight worsening of the loss is probably due to the fact that, unlike the experiment done on the MVTec AD benchmark dataset, this industrial dataset presents many more difficulties. This is because it is representative of a real use case. The sum of the OD part to the cross entropy leads to this slight deterioration. This phenomenon does not occur on the MVTec AD dataset because, being simpler, the network

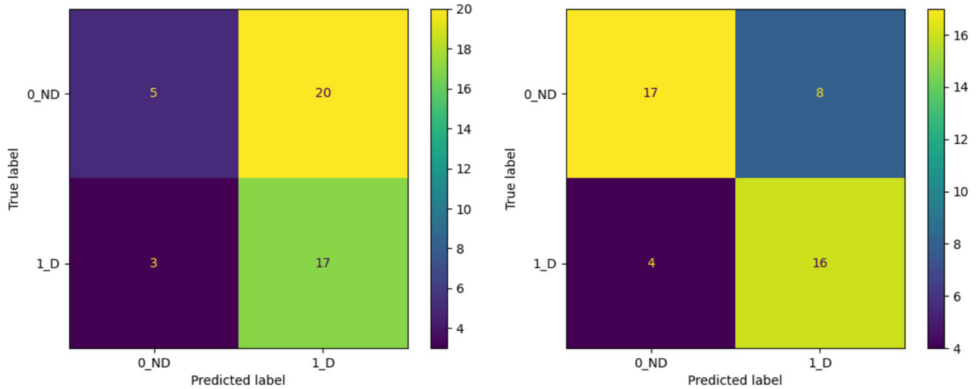
Table 1 MVTec AD Dataset.*CE* and *Exp.* are the acronyms for *Cross-Entropy loss* and *experiment* (in bold the best results)

CNN	Exp.	Training			validation		
		Accuracy	Loss	OD	Accuracy	Loss	OD
EfficientNet-B0	CE	1.000	0.0049	-	0.992	0.022	-
	CEOD	0.990	0.005	0.0005	**1.00**	**0.010**	0.00032
CustomNet	CE	0.999	0.008	-	0.910	0.35	-
	CEOD	0.998	0.010	0.0002	**0.960**	**0.09**	0.0003



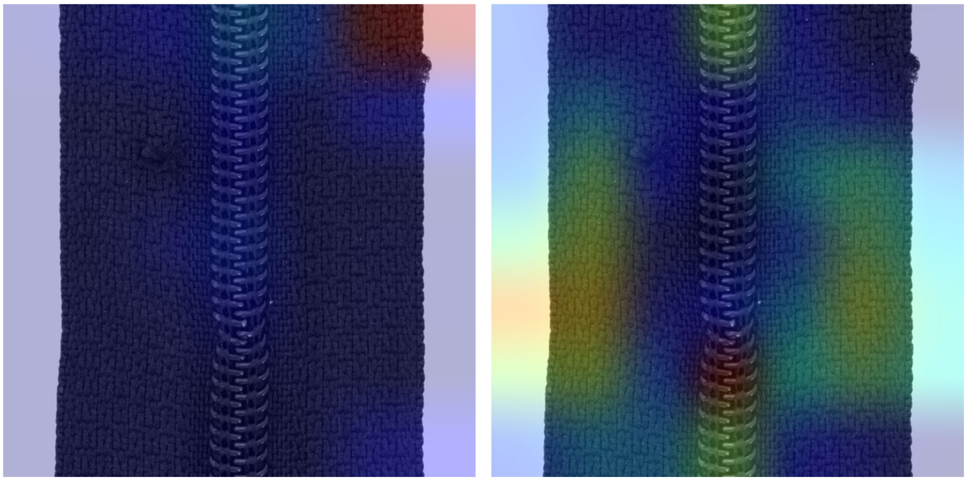
(a) CE (b) CEOD

Fig. 10 Confusion matrices on test set of MVTec AD dataset obtained with EfficientNet-B0. *0_ND* and *1_D* represent the class without and with defects respectively



(a) CE (b) CEOD

Fig. 11 Confusion matrices on test set of MVTec AD dataset obtained with custom CNN. *0_ND* and *1_D* represent the class without and with defects respectively

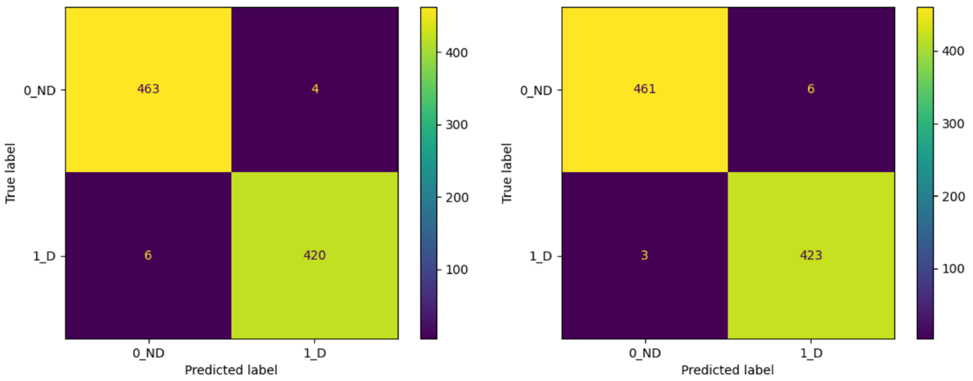


(a) CE (b) CEOD

Fig. 12 In (a) there is the heatmap produced with the classification with standard Cross-Entropy loss (CE) network. It is possible to see that the defect on the zip is not highlighted, unlike the external defect in the upper right. In (b) there is the heatmap produced by the network trained with CEOD. In this case, the defect on the zip is highlighted, unlike external defects which are not considered by the network

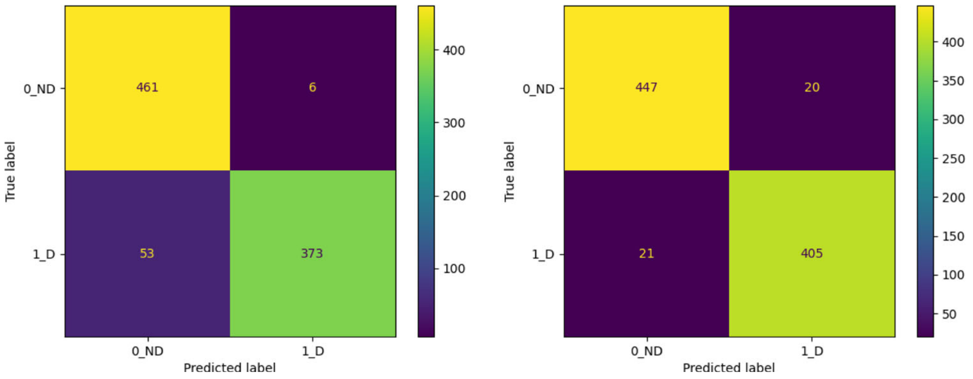
Table 2 Industrial Dataset. For EfficientNet-B0, results for Transfer Learning (TL) and fine-tuning (ft) (in bold the best results)

CNN	Exp.	Training			validation		
		Accuracy	Loss	OD	Accuracy	Loss	OD
EfficientNet-B0 - TL	CE	0.956	0.11	-	0.976	0.102	-
	CEOD	0.957	0.10	0.00055	**0.977**	**0.110**	0.00036
EfficientNet-B0 - ft	CE	0.990	0.00018	-	0.995	0.017	-
	CEOD	1.000	0.0007	0.00038	**0.998**	**0.045**	0.00032
CustomNet - ft	CE	1.000	0.00001	-	0.98	0.053	-
	CEOD	1.000	0.00002	0.000017	**0.9965**	**0.013**	0.00002



(a) CE (b) CEOD

Fig. 13 Confusion matrices on test set of the industrial dataset obtained with EfficientNet-B0. *0_ND* and *1_D* represent the class without and with defects respectively



(a) CE (b) CEOD

Fig. 14 Confusion matrices on test set of the industrial dataset obtained with the custom CNN. *0_ND* and *1_D* represent the class without and with defects respectively

trained with CEOD can clearly exceed that trained with the standard loss and therefore the effect of the sum of the ODs does not lead to worsening the loss. However, despite the excellent results obtained through the network trained for standard classification, with CEOD we are able to obtain a further increase in performance. Figures 13 and 14 show the results of EfficientNet-B0 and the custom CNN on the *test set* of the industrial dataset. In Figures 13 and 14, we can see that the networks trained with CEOD obtain better results in terms of defect identification at the expense of a slight worsening in the identification of non-defective ones. EfficientNet-B0 trained with CEOD reach 98.9% accuracy and 0.99 of AUCROC compared to EfficientNet-B0 trained with standard cross-entropy that reaches 98.8% accuracy and 0.98 of AUCROC on the test set. The custom CNN trained with CEOD reaches 95.4% accuracy and 0.95 of AUCAUC as compared to the CNN trained with the standard loss that reaches 93.3% accuracy and 0.93 of AUCROC on the test set. The time spent performing 360 training cycles with EfficientNet-B0 trained with standard cross-entropy is 9h 25m 31s. The time spent performing 360 training cycles with EfficientNet-B0 trained with CEOD is 8h 12m 47s. The time spent with the custom CNN to perform 80 training cycles with standard loss is 1h 44m 25s versus 1h 44m 5s for the custom CNN trained with CEOD.

6 Conclusions

The aim of this work is to improve the use of a CNN in the field of anomaly detection by stimulating the network to pay attention mainly to a specific part of an image, to avoid the identification of part of images containing noise defects in the background. This work presents a new loss that acts as an attention mechanism to make a neural network focus on a specific part of an image (called Area of Interest - AOI). This area might not even be the same along all the datasets. This goal of our work was achieved by extending the Szymkiewicz - Simpson Overlap coefficient to obtain what we have defined *Overlap Distance* (OD). All these contributions were added to the loss function used for the classification task (cross-entropy loss). The experiments show that our approach performs better than standard cross-entropy on both benchmark and industrial real-case datasets.

The introduction of the Overlap Distance (OD) as an attention mechanism represents a significant advancement in anomaly detection using convolutional neural networks. By focusing the network's attention on specific Areas of Interest (AOI), we mitigate the risk of false positives caused by background noise defects. This innovation holds promise not only in image processing but also in various domains where precise attention allocation is crucial.

The next step for this project is to try to apply this new type of loss to an unsupervised learning framework. We are studying a way to implement this approach into GANs neural network for anomaly detection. This is motivated by the high suitability of GANs for the anomaly detection task in the industrial sector.

The potential impact of this research extends beyond anomaly detection, with implications for a range of industries reliant on accurate image analysis and pattern recognition. We believe that these advancements will contribute to more robust and reliable quality control processes in the industrial sector.

Acknowledgements The authors want to thank the Italian company for providing a real-use case dataset to test the software developed in this work. The first author is supported by a PhD scholarship funded by Emilia- Romagna region, under POR FSE 2014-2020 program. The authors also acknowledge "SUPER: Supercomputing Unified Platform - Emilia-Romagna" project, financed under POR FESR 2014-2020. This work was partly supported by the "National Group of Computing Science (GNCSINDAM)".

Funding Open access funding provided by Università degli Studi di Ferrara within the CRUI-CARE Agreement.

Data Availability The MVTec AD dataset analysed during the current study are available at https://www.mvtec.com/company/research/datasets/mvtec-ad. The Industrial datasets analysed during the current study are not publicly available due to NDA signed with the company that provided the dataset.

Declarations

Conflicts of interest The authors declare no competing interests.

References

1. Zheng X, Zheng S, Kong Y, Chen J (2021) Recent advances in surface defect inspection of industrial products using deep learning techniques. The International Journal of Advanced Manufacturing Technology. 113(1):35–58. https://doi.org/10.1007/s00170-021-06592-8
2. Liu G, Yang N, Guo L, Guo S, Chen Z (2020) A one-stage approach for surface anomaly detection with background suppression strategies. Sensors. 20(7):1829

3. MarkouM, Singh S (2004) Novelty detection: a reviewpart 2: statistical approaches
4. Hodge V (2004) Austin J: A survey of outlier detection methodologies. Artif Intell Rev
5. Weimer D, Shpitalni M, Scholz-Reiter B (2016) Design of deep convolutional neural network architectures for automated feature extraction in industrial inspection. CIRP
6. Staar B, Lütjen M, Freitag M (2019) Anomaly detection with convolutional neural networks for industrial surface inspection. Procedia CIRP
7. Hoffer E, Ailon N (2015) Deep metric learning using triplet network. International Workshop on Similarity-Based Pattern Recognition
8. Akcay S, Atapour -Abarghouei A, Breckon TP (2018) Ganomaly: Semi-supervised anomaly detection via adversarial training. Asian conference on computer vision. Springer, pp 622–637
9. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y (2014) Generative adversarial nets. Adv Neural Inf Process Syst 27
10. Jinwon A, Sungzoon C (2015) Variational autoencoder based anomaly detection using reconstruction probability. Special Lect IE 2(1):1–18
11. Kingma D, Welling M (2014) Auto-encoding variational bayes. International Conference on Learning Representation
12. Ferrari N, Fraccaroli M, Lamma E (2023) Grd-net: Generative-reconstructive-discriminative anomaly detection with region of interest attention module. Int J Intell Syst 2023:7773481. https://doi.org/10.1155/2023/7773481
13. Moon IY, Lee HW, Kim S-J, Oh Y-S, Jung J, Kang S-H (2021) Analysis of the region of interest according to cnn structure in hierarchical pattern surface inspection using cam. Materials. 14(9):2095
14. Cohen N, Hoshen Y (2020) Sub-image anomaly detection with deep pyramid correspondences. arXiv:2005.02357
15. Yi J, Yoon S (2020) Patch svdd: Patch-level svdd for anomaly detection and segmentation. In: Proceedings of the Asian Conference on Computer Vision
16. Kimura D, Chaudhury S, Narita M, Munawar A, Tachibana R (2020) Adversarial discriminative attention for robust anomaly detection. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. pp 2172–2181
17. Venkataramanan S, Peng K-C, Singh RV, Mahalanobis A (2020) Attention guided anomaly localization in images. In: European conference on computer vision. Springer, pp 485–503
18. Song J, Kong K, Park Y-I, Kim S-G, Kang S-J (2021) Anoseg: Anomaly segmentation network using self-supervised learning. arXiv:2110.03396
19. Selvaraju RR, Cogswell M, Das A, Vedantam R, Parikh D, Batra D (2017) Grad-cam: Visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE international conference on computer vision. pp. 618–626
20. Xu F, Wang H, Sun X, Fu X (2022) Refined marine object detector with attention-based spatial pyramid pooling networks and bidirectional feature fusion strategy. Neural Comput Appl 34(17):14881–14894
21. Xu F, Wang H, Peng J, Fu X (2021) Scale-aware feature pyramid architecture for marine object detection. Neural Comput Appl 33:3637–3653
22. Wang H, Peng J, Zhao Y, Fu X (2020) Multi-path deep cnns for fine-grained car recognition. IEEE Trans Veh Technol 69(10):10484–10493
23. Zhou A, Ai B, Qu P, Shao W (2021) Defect detection for highly reflective rotary surfaces: An overview. Meas Sci Technol 32(6):062001
24. Bergmann P, Batzner K, Fauser M, Sattlegger D, Steger C (2021) The mvtec anomaly detection dataset: a comprehensive real-world dataset for unsupervised anomaly detection. Int J Comput Vision 129(4):1038–1059
25. Bergmann P, Fauser M, Sattlegger D, Steger C (2019) Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9592–9600
26. Vijaymeena MK, Kavitha K (2016) A survey on similarity measures in text mining
27. Zhou B, Khosla A, Lapedriza A, Oliva A, Torralba A (2016) Learning deep features for discriminative localization. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2921–2929
28. He K, Gkioxari G, Dollár P, Girshick R (2018) Mask R-CNN
29. Chen L-C, Papandreou G, Kokkinos I, Murphy K, Yuille AL (2017) Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE Trans Pattern Anal Mach Intell 40(4):834–848
30. Ronneberger O, Fischer P, Brox T (2015) U-net: Convolutional networks for biomedical image segmentation. International conference on medical image computing and computer-assisted intervention. Springer, pp 234–241
31. Mannor S, Peleg D, Rubinstein R (2005) The cross entropy method for classification. In: Proceedings of the 22nd international conference on machine learning. pp 561–568
32. Tan M, Le Q (2019) Efficientnet: Rethinking model scaling for convolutional neural networks. International Conference on Machine Learning. PMLR, pp 6105–6114
33. Deng J, Dong W, Socher R, Li L-J, Li K, Fei-Fei L (2009) Imagenet: A large-scale hierarchical image database. 2009 IEEE conference on computer vision and pattern recognition. Ieee, pp 248–255
34. Bagherinezhad H, Horton M, Rastegari M, Farhadi A (2018) Label refinery: Improving imagenet classification through label progression. arXiv:1805.02641
35. Yang S, Xiao W, Zhang M, Guo S, Zhao J, Shen F (2022) Image data augmentation for deep learning: A survey. arXiv:2204.08610
36. Shorten C, Khoshgoftaar TM (2019) A survey on image data augmentation for deep learning. J Big Data 6(1):1–48

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Michele Fraccaroli is a software engineer who works at a company that develops solutions for fruit selection and processing. He obtained his Ph.D. at the Department of Engineering, University of Ferrara. He graduated in Computer Science and Automation Engineering at the Department of Engineering, University of Ferrara, in 2019. His research is mainly focused on deep learning, in particular on neural-symbolic integration systems and eXplainable Artificial Intelligence (XAI). He also deals with issues relating to industrial visual inspection, deep and machine learning applied to medicine and computer vision.

Alice Bizzarri is a Ph.D. student in Artificial Intelligence at the Department of Engineering of the University of Ferrara. She received her degree in Computer and Automation Engineering from University of Ferrara Italy, in March 2021.

Paolo Casellati has a master degree in Computer Engineering from University of Ferrara. ly works as a software designer engineer for a factory specialized in precision measuring instruments.

Evelina Lamma is Full Professor of Computer Science - Artificial Intelligence at the Department of Engineering of the University of Ferrara. She received her degree in Electronic Engineering from University of Bologna, Italy, in 1985 and her Ph.D. degree in Computer Science in 1990. Her research activity focuses around artificial intelligence, knowledge representation, medical imaging and computer vision, computational logic, data mining and machine learning.