Journal of Ambient Intelligence and Humanized Computing
https://doi.org/10.1007/s12652-021-03688-7

ORIGINAL RESEARCH

Industrial object and defect recognition utilizing multilevel feature extraction from industrial scenes with Deep Learning approach

Ioannis D. Apostolopoulos[1] · **Mpesiana A. Tzani**[2]

Received: 7 December 2020 / Accepted: 21 December 2021

Abstract
Modern industry requires modern solutions for monitoring the automatic production of goods and detecting defected materials. Smart monitoring of the functionality of the mechanical parts of technology systems or machines is a mandatory step towards automatic production. Deep Learning has proven its efficiency in feature extraction from images, videos and text, thereby succeeding in various object detection, recognition, segmentation and classification tasks. Despite its advances, little has been investigated about the effectiveness of specially designed Convolutional Neural Networks (CNNs) for defect detection and industrial object recognition. In the particular study, we employed six publicly available industrial-related image datasets, containing defected materials and industrial tools, or engine parts, aiming to develop a specialized model to classify them. Motivated by the success of the Virtual Geometry Group (VGG) network, we propose a modified version of it, called Multipath VGG19, which allows for extra local and global feature extraction (multi-level feature extraction) by making use of several processing paths. The extra features are fused via concatenation. The experiments verified the effectiveness of MVGG19 over the baseline VGG19. Specifically, top classification performance was achieved in five of the six image datasets, whilst the average classification improvement was 6.95%. MVGG19 also showed better overall stability and robustness to dataset variation, compared to other baseline state-of-the-art CNNs.

Keywords Deep Learning · Machine part recognition · Defect detection · Industrial object recognition · Production monitoring

1 Introduction

Modern production automation systems have opened up great horizons simplifying many functions of the production process, accelerating the production, maintenance, and transportation of products. The introduction of several automation systems requires the corresponding automatic control for the timely and the valid detection of errors, the confrontation of dangerous situations, and the smooth maintenance of the machines (Diez-Olivan et al. 2019). Such procedures are required to be carried out in real-time with the utilization of appropriate equipment. Hence, humanity aims to convert the production process into a smart one.

In recent years, image and video based object recognition has been one of the most actively researched Artificial Intelligence (AI) tasks (Han et al. 2018). It refers to technologies that, through specific algorithms based on Machine Learning (ML), can classify specifically targeted subjects. Accordingly, object recognition is a computer vision technique used to recognize and detect objects within an image or video (Khan et al. 2018). Object recognition consists of recognizing, identifying, and locating objects within an image with a given amount of confidence. Specifically, object detection draws bounding boxes around these detected objects to identify where the objects are in (or how they pass through) a particular scene.

In several industry sectors, object recognition has many exciting uses. For example, automated systems can be built to recognize defective parts or tools for immediate replacement and also to detect individual parts that require refinement and/or replacement during the manufacturing process. Even the ability to scan for objects and measure their number in images is essential for a company,

✉ Ioannis D. Apostolopoulos
ece7216@upnet.gr

[1] Department of Medical Physics, School of Medicine, University of Patras, 265-00 Patras, Greece

[2] Department of Electrical and Computer Engineering, University of Patras, 265-00 Patras, Greece

Published online: 08 January 2022

particularly in the industry. The regular tasks of manually measuring the number of different components or objects are an essential part of the working time of the specialists. The application of research in the field of deep Convolutional Neural Networks (CNN) to the tasks of detection and classification can also help to automate specific repetitive tasks.

The automation of such tasks involves deep analysis of the input data and massive feature extraction to define the decisive characteristics that define an object.

Recently, object recognition from digital images or videos made significant progress thanks to the development of new image processing techniques also known as Deep Learning (DL) with Convolutional Neural Networks (Yan et al. 2015; Goodfellow et al. 2016). Deep Learning alludes to various ML methods and has already succeeded in speech and image recognition, natural language processing, and more (Najafabadi et al. 2015; Apostolopoulos et al. 2021). CNNs are a special type of the traditional Neural Networks, which employ the convolution process to analyze the input data distributions and generate potentially powerful features related to a specific domain.

ML and DL brought a revolution in feature extraction from any input data. Manual feature extraction from images, for example, was constrained to pre-defined features (e.g. the color, the diameter and the texture of an object). Manual feature extraction still succeeds in a variety of tasks (Wagner et al. 2018; Khan and Yong 2016; Zeng et al. 2018). The innovation of ML and DL lies in the automatic extraction of millions of features and their classification based on their importance for the desired task. It is undeniable that a large proportion of the extracted features may be irrelevant, misleading, or overlapping with each other. This is why ML and DL approaches also aim to distinguish the most important ones.

Deep Learning (DL) could be extremely useful in industrial projects. In recent years, the research community has put particular emphasis on developing such image processing systems for materials in manufacturing as a way of promoting all sorts of functions that make production time-consuming and expensive.

Motivated by the success of the DL model called VGG19 (Virtual Geometry Group) (Simonyan and Zisserman 2014), we propose a VGG-based DL framework capable of recognizing multiple defects and objects from multiple image sources.

The major difference of the proposed model from the baseline VGG lies in the feature extraction procedure. VGG and its derivatives (VGG16, VGG19) process the input image in a sequential manner. The input scene is subject to many sequential convolutions, which extract millions of features. However, each convolution is applied to the previous product of another convolution. In this way, important low-level features may be distorted as we go deeper into the network. The modification this study proposes aims to extract more features (both low and high level) by making use of multiple feature extraction paths. The extracted features are fused before the classification, by simply concatenating the feature maps.

The contributions of this study could be summarized as follows:

a) A uniform and effective VGG19-based CNN is proposed, which achieves effective feature extraction from industrial-type scenes.
b) The model's evaluation is based on six industrial image datasets and indicates that the model is successful and superior to the baseline VGG and other state-of-the-art CNNs, indicating that it could stand out as a general model for feature extraction and image classification of industrially related scenes.

2 Related work

The efficacy of DCNNs in various scenarios of manufacturing (inspection, motion detection, and more) has been demonstrated in recent years (Gu et al. 2017, 2018b; Wang et al. 2018a).

Caggiano et al. (2019) developed a ML method based on Deep Convolutional Neural Network (DCNN) to detect defects based on SLM non-compliance through automated image processing. In particular, a ML method has been developed for the online detection of defects through automated image processing, leading to the timely detection of defective sections of a material due to non-compliance with the Selective Laser Melting (SLM) metal powder process. During layer-by-layer SLM processing, the images in this study were obtained and the analysis was performed using a Deep Convolutional Neural Network model based on two currents and automatic image learning and feature fusion achieved the identification of the defective condition-related SLM pattern.

The work of Fu et al. (2019), which focuses on automated visual identification of steel surface defects, is impressive and can make a major contribution to functionality in order to facilitate quality control of the output of steel strips. In order to achieve fast and accurate classification of defective steel surfaces, the authors present an effective model of a CNN, which emphasizes the training of low-level features and incorporates multiple receptive fields. Their methodology focuses on three basic modules, first on the use as a fundamental architecture of pre-trained SqueezeNet (Iandola et al. 2016). Second, in a series of reinforced diversity surface steel defect data containing extreme non-uniform lighting, camera noise and motion blur, the use of only a

limited amount of defect-specific training samples for high-accuracy detection. Finally, by running over 100 fps on a machine equipped with a single NVIDIA TITAN X graphics processing unit (12G memory), the lightweight CNN model they used will fulfill the requirement for real-time online inspection.

Another approach from Wang et al. (2019) refers to a new mechanical vision inspection system focused on learning to identify and classify a faulty product without loss of precision. The Gauss filter is used explicitly in this work. The contribution of this particular work lies in the unloading of the computational burden of the next identification process because of the export of a region of interest (ROI) based on the transformation of Hough to eliminate the irrelevant context. In order to achieve a good balance between detection accuracy and computational performance, the construction of the recognition unit is based on a convergent neural network, while the remaining inverted block is implemented as the basic block. By using the proposed approach with a large number of data sets consisting of inaccurate and defective bottle images, superior control efficiency is achieved. The authors emphasize that the monitoring system is capable of covering both precision and effectiveness by combining traditional methods of image processing and a light deep neural network.

Despite the remarkable progress and the numerous DL proposals for object and defect recognition, little has been said about the effectiveness of universal models, specifically designed for images with industrial content, such as images illustrating machines, metallic parts, tools, etc. In this context, the present research study proposes a DL model to recognize several industry-related objects and defects from a variety of domains.

3 Material and methods

The main advantage of CNNs lies in extracting new features from the input data distributions (i.e., images, videos), thereby bypassing the manual feature extraction process, which is traditionally performed in image analysis tasks with ML methods (Lin et al. 2017).

Each convolution layer in a CNN is processing the previous layer's output by applying new filters and extracting new features (LeCun et al. 2015). Since the convolutional layers are stacked together, a hierarchical process takes place. In essence, features directly from the original image are only extracted by the first convolutional layer, whereas the other layers process each other's outputs. In this way, a slow introduction to large amounts of filters is achieved, while underlying features may be revealed during the last layers. The general rule of thumb relates the network's effectiveness with the number of convolutional layers (Shin et al. 2016). This is why deep networks are generally superior, provided that an adequate amount of image data is present. In cases where the dataset's size is not large enough to feed a deep network, three solutions are commonly proposed:

a) The selection of a simpler CNN, which contains fewer trainable parameters and fits in the particular data well
b) Transfer learning (Kornblith et al. 2019), utilizing deep and complex CNNs but freezing their layers, thereby decreasing the trainable parameters and allowing for knowledge transfer, following their training on large image datasets.
c) Data augmentation (Wong et al. 2016) methods to increase the training set size.

In the particular study, although the size of the training data is not negligible, we propose both transfer learning and data augmentation to increase the training set further and train a robust model with the ability to generalize.

3.1 Transfer learning

Transfer learning (Kornblith et al. 2019), refers to the procedure wherein a ML or DL model developed for a certain task is reused for a second task. Usually, transfer learning involves tuning the parameters and the hyper parameters of the transferred model, to ensure best fit in the new data (Weiss et al. 2016).

The most naïve transfer learning takes place when the model's architecture and the learned weights are retained and are directly applied to process the images of the desired task (Zhuang et al. 2020). The reader should note that the weights are learned by its initial training and not by training from scratch on the data of the second task. In this way, the model acts as feature extractor, i.e., the model seeks and extracts old-task-related features and is not learning to extract new-task-specific ones. It is common to connect a trainable Neural Network at the top of the CNN, to perform the classification and learn to distinguish the important features from the irrelevant.

Advanced transfer learning methods involve maintaining the original CNN structure, but not all the learned weights. In this process, a proportion of the CNN's layers may discard their weights and learn new, when trained with images of the second task. Several experiments may be required to define the exact amount of layers to be trainable. Transfer learning may also involve modifications to the initial architecture of the CNN.

Transfer Learning may be employed for several reasons: (a) data scarcity (for the second task), i.e., the available data are too few to train all the layers of a model, (b) the source task and the target task are similar, (c) reduction of computational cost.

3.2 Data augmentation techniques

Data augmentation is an essential step in DL applications and research, mainly utilized for two reasons (Fawzi et al. 2016). The first reason is the data scarcity, which impedes DL models adaptation to the domain of interest and develop their learning capabilities. Few images are commonly not enough for a DL framework to train on. Especially in cases where the classification should be based on deep features and not obvious and low-level characteristics (e.g., colors). With data augmentation, the initial training set can be broadly expanded by applying various transformations to the original images. In this way, the model learns to ignore useless characteristics and improves its spatial capabilities. For example, applying random rotations directs the model towards seeking patterns in moving positions and not fixed.

By data scarcity, not only is the shortage of data considered, but also the shortage of data covering the diversity of the statistical population of the target classification. For example, a dataset may contain thousands of images illustrating the front of a car, based upon which the developed model shall distinguish between old-fashioned and modern cars. Supplying a new, unseen image to the model for prediction, where the rear of a random car is depicted, may lead the model to the wrong conclusion since the initial training had been made utilizing solely front-view images. Data augmentation may modify or generate completely new images, capturing essential features found in both front-view and rear-view images, thereby expanding the training sets' diversity (Zoph et al. 2020). This is the second reason why data augmentation is preferable.

In the present research, we applied the following augmentations to the training sets to expand the available data and to increase the generalization capabilities of the experimental DL networks:

a. Random Gaussian Noise to increase the irrelevant artifacts in the image and help the model focus on real and decisive features, rather than irrelevant information
b. Random Rotations to ensure spatial exploration for features
c. Horizontal and Vertical Flips to ensure that the model does not distinguish objects based-n their orientation
d. Height and Width shifts to ensure that the model does not distinguish objects based on their location in the image.

The reader should note that data augmentation is applied solely to the training sets and not the test sets. Data augmentations have been applied using the *Image Data Generator class* of the Keras library.

3.3 Multipath VGG19

Because the low-level features, which are generally extracted from the first layers of any sequential CNN, may be useful for the classification task (Yue-Hei Ng et al. 2015), a completely hierarchical network may lack on this front due to the following: Typical CNNs fuse the initially extracted features with deeper features because one convolution comes after the other. To circumvent this setback, we investigate and experiment with a modification of the original structure of the CNN to allow for parallel feature extraction, which is achieved utilizing many paths.

Inspired by the Victual Geometry Group network (Simonyan and Zisserman 2014), in this study, a novel modification is proposed and analyzed. The baseline VGG19 is a uniform and straightforward CNN, consisting of five groups of convolutions of 64, 128, 256, 512, and 512 depth (filters). Each convolution group is followed by the Max Pooling operation to achieve dimensionality reduction. VGG19's architecture makes it suitable for modifications, due to its simplicity, uniformness, and transparency.

Multipath VGG19, as shown in Fig. 1, takes full advantage of the traditional hierarchical structure of VGG19 while it connects early and late convolutional blocks together by constructing extra feature map processing paths. After the second, third, and fourth max-pooling layers, the extracted feature maps are processed by "BD" layers, which apply batch normalization, dropout, and global average pooling. The three BD outputs and the sequential output are concatenated and fed to a neural network of 2500 neurons.

The idea behind this modification is to disconnect the early and late extracted features from the sequential structure and connect them directly to the classifier at the top of the network. In this way, the total extracted features are increased and each path is responsible for transferring features from the early and the late image process that takes place. Therefore, the BD blocks are not adding any other convolution operation, but are just transferring, normalizing and reducing the dimensions of them.

Each BD box produces feature maps in various sizes. For example, the first BD block has an output of (50, 50, 256), where 50, 50 refer to the width and the height of the feature map (2D image, practically) and the number 256 refers to the 256 sets of 2D feature maps being produced. The rest BD boxes' outputs are of different size. To achieve a uniform concatenation of those features, the Global Average Pooling of each BD block ensures that the feature maps are encoded to a one-sized array (e.g. of 512 size). In this way, the feature maps are transformed from 3 to 1D, thereby making their fusion simple. The features are fused by utilizing the concatenation function (Du et al. 2020), which stacks the features together.

A softmax classifier performs the feature classification according to the desired number of classes (in Fig. 1, two

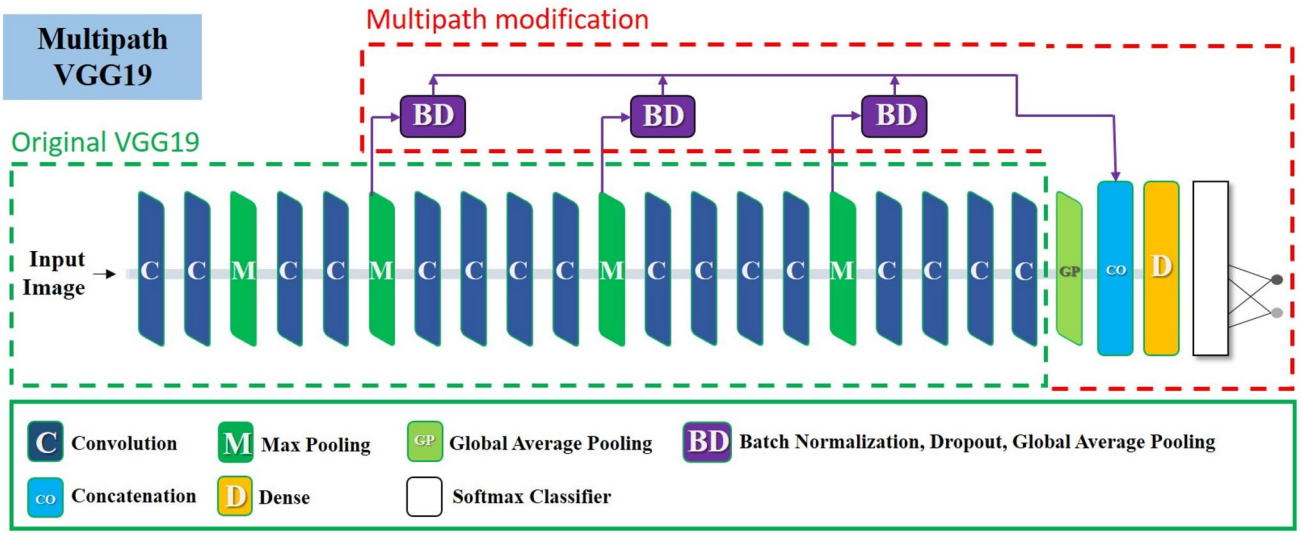

Fig. 1 Multipath VGG19

Table 1 MVGG19 parameters

MVGG19 parameters	Selection
Trainable layers	The final convolution layer
Batch normalization	Across the entire network
Dropout	In BD blocks (50% dropout)
Global pooling	Average pooling
Pooling between convolutions	Max pooling
Classifier at the top	Neural Network (2500 nodes)
Optimization	Adam
Batch size for training	Depends on the dataset's size, typical value = 64
Epochs of training	Depends on the dataset, typical value = 30–60 epochs

arbitrary classes are shown). The source code for constructing MVGG19 from scratch is provided in *Github repository*.

MVGG19 is fine-tuned (Tajbakhsh et al. 2016) by maintaining the top (i.e., the last) layer trainable, while the rest of the layers are frozen to retain their weights obtained from its initial training, which was conducted using images from the ImageNet challenge (Krizhevsky et al. 2017). In this way, VGG19 and MVGG19 come prepared for the feature extraction by retaining their learned weights (i.e. knowledge) obtained by a more complete training for image classification, that of ImageNet challenge.

We used batch normalization and 50% dropout after the dense layer of 2500 neurons and inside the BD blocks. Dropout (Poernomo and Kang 2018) is a simple, yet effective technique to prevent overfitting, i.e. to prevent the network from learning too specific information and depending its predictions on features captured in a specific image and do not represent global features that describe the desired object.

Optimization was achieved employing the default parameters of the Adam (Kingma and Ba 2014) optimization algorithm. We selected Adam due to its success in optimization of baseline VGG19 in a variety of similar tests (Mehta et al. 2019; Deitsch et al. 2019).

Information regarding the structure of MVGG19 are given in Supplementary Material.Parameters and hyper parameters of the network are presented in Table 1. MVGG19 parameters.

All experiments were performed in a python environment making use of the Keras library. An Intel Core i5-9400F CPU at 2.90 GHz computer equipped with 6 Gb RAM and a GeForce RTX 2060 Super was the primary infrastructure for the experiments.

3.4 Image datasets for industrial object recognition and defect detection

For the particular study, we selected six publicly available sets of images related to industrial applications. The characteristics of the evaluation datasets are described in this section.

3.4.1 Industrial dataset of casting production (casting dataset)

This dataset is of casting manufacturing product. Casting is a manufacturing process in which a liquid material is usually poured into a mould, which contains a hollow cavity of the desired shape, and then allowed to solidify. All included images are top views of the submersible pump impeller. The total images are 7348, while their size is 300×300 pixels. Two categories are describing the contents of each image, namely defective object and normal. The images were very well organized, and no image preprocessing was required. The dataset is openly available in Kaggle (Dabhi 2020). Examples of the two classes are depicted in Supplementary Material. The task of the proposed MVGG19 model is to correctly detect the problematic parts.

3.4.2 Defects location for metal surface dataset (Defect dataset)

Known initially as GC10-DET dataset (Lv et al. 2020), this set refers to ten types of surface defects, i.e., punching (Pu), weld line (Wl), crescent gap (Cg), water spot (Ws), oil spot (Os), silk spot (Ss), inclusion (In), rolled pit (Rp), crease (Cr), waist folding (Wf). The collected defects are on the surface of the steel sheet. The dataset includes 3570 Gy-scale images. Distinguishing between those types of defection is crucial for the industry. It can significantly contribute to the prevention of malfunction, the real-time identification of defects on a variety of essential materials found in factories, processing plants, and logistics. The dataset is publicly available at Github. In Supplementary Material, the reader can observe samples from this dataset.

3.4.3 Magnetic tile defect dataset (magnetic tile dataset)

The present dataset (Huang et al. 2018) is available for research purposes in a variety of repositories. The images of six common magnetic tile defects were collected, while the dataset contains annotations for segmentation tasks. The dataset contains 1243 images of those 6 classes. Samples are illustrated in Supplementary Material.

3.4.4 Object recognition in industry dataset (Tech dataset)

Focused on industrial applications, The MVTec Industrial 3D Object Detection Dataset (MVTec ITODD) is a public dataset for object detection and pose estimation in 2D or 3D (Drost et al. 2017). It consists of 28 object classes and more than 3500 labeled images of those objects. For the particular task, 10 object types are selected, namely cylinder, planar bracket, star, fuse, box, round engine cooler, cap, engine cover, car rim, and bracket screw. Samples of those objects are illustrated in Supplementary Material.

3.4.5 Bridge crack recognition dataset (bridge dataset)

This dataset's original name is SDNET2018 (Maguire et al. 2018), which is a publicly available annotated dataset intended for the evaluation of artificial intelligence algorithms. It contains over 56 thousand images of cracked and non-cracked concrete bridge decks, walls, and pavements, where cracks are as narrow as 0.06 mm and as wide as 25 mm. Shadows, surface roughness, scaling, edges, holes, and background debris are artifacts included in the images, making the recognition task more challenging. Two classes are generated from this dataset, namely "cracked" and "ok."

3.4.6 Solar Cell defect probability dataset (solar cell defect (ELPV) dataset)

Solar Cell defect probability dataset was acquired from Github (Buerhop-Lutz et al. 2018; Deitsch et al. 2019; Deitsch et al. 2020). It contains functional and defective solar cell surfaces with a variety of degradation degrees. For the particular experiment, two classes are created, namely "defect" and "ok", where images annotated with a degree above zero are considered defective. As mentioned in the repository, all the included images had been normalized with respect to their size and their perspective. Prior to solar cell extraction, the distortions induced by the camera utilized has been eliminated. The overall size of the dataset is 2624 images, while each image size is 300×300 pixels of 8-bit. 44 different solar modules were examined (Table 2).

One limitation of this study is the imbalance between defected-materials datasets and actual object recognition datasets. However, by the time this study was conducted, there was a shortage of available image data that could be included.

4 Results

The epochs of training and the batch size were adjusted for each dataset to perform an optimal training fitting to the particular computational infrastructure. For the evaluation

Table 2 Overview of the datasets

Dataset name	Total number of images	Classes
Casting	8648	2
Defect	2306	10
Magnetic tile	1243	6
Tech	2349	27
Bridge	54,999	2
Solar Cell defect	2624	2

Table 3 MVGG19 results

Dataset	Accuracy (%)	Precision (%)	Recall (%)	F-1 (%)	AUC (%)
Casting	97.88	96.49	98.57	97.5	99.59
Defect	77.62	77.68	–	–	94.94
Magnetic Tile	92.67	98.49	–	–	97.61
Tech	94.23	98.13	–	–	99.94
Bridge	99.02	99.5	99.36	99.43	99.83
Solar (ELPV)	76.78	75.63	67.02	70.93	83.36

F-1 F1 score; *AUC* Area Under Curve score

of the performance, tenfold cross-validation was preferred. During this process, ten independent training–testing phases took place. For each phase, a different part of the dataset was selected to serve as a test (i.e., hidden) set, while the rest of the dataset was utilized for training. The results are aggregated, and the performance metrics correspond to the mean of the metrics recorded during each phase.

4.1 Multipath VGG19

For the majority of the datasets, the proposed network achieves top accuracy and Area Under Curve score. Specifically, the best classification accuracy is obtained for the Bridge dataset, with 99.02%. The second-best accuracy obtained is 97.88% in the Defect dataset. The rest of the accuracy results are 94.23% for the Tech dataset, 92.67% for the Magnetic dataset, 77.62% for the Casting dataset, and 76.68% for the Solar dataset. Table 3 presents the analytical results of MVGG19 for each of the datasets.

Due to limitations of size, two figures presenting the best and the worst results are illustrated. The rest of the images can be found in the supplementary material. In Fig. 2, the results for the Casting dataset are presented. In Fig. 3, the results for the Bridge dataset are presented. In Fig. 2, sub-figure b), epoch 5 seems to yield abnormal validation loss. This could be explained by the fact that until the training is complete, the learned weights are not adequate to correctly predict unseen

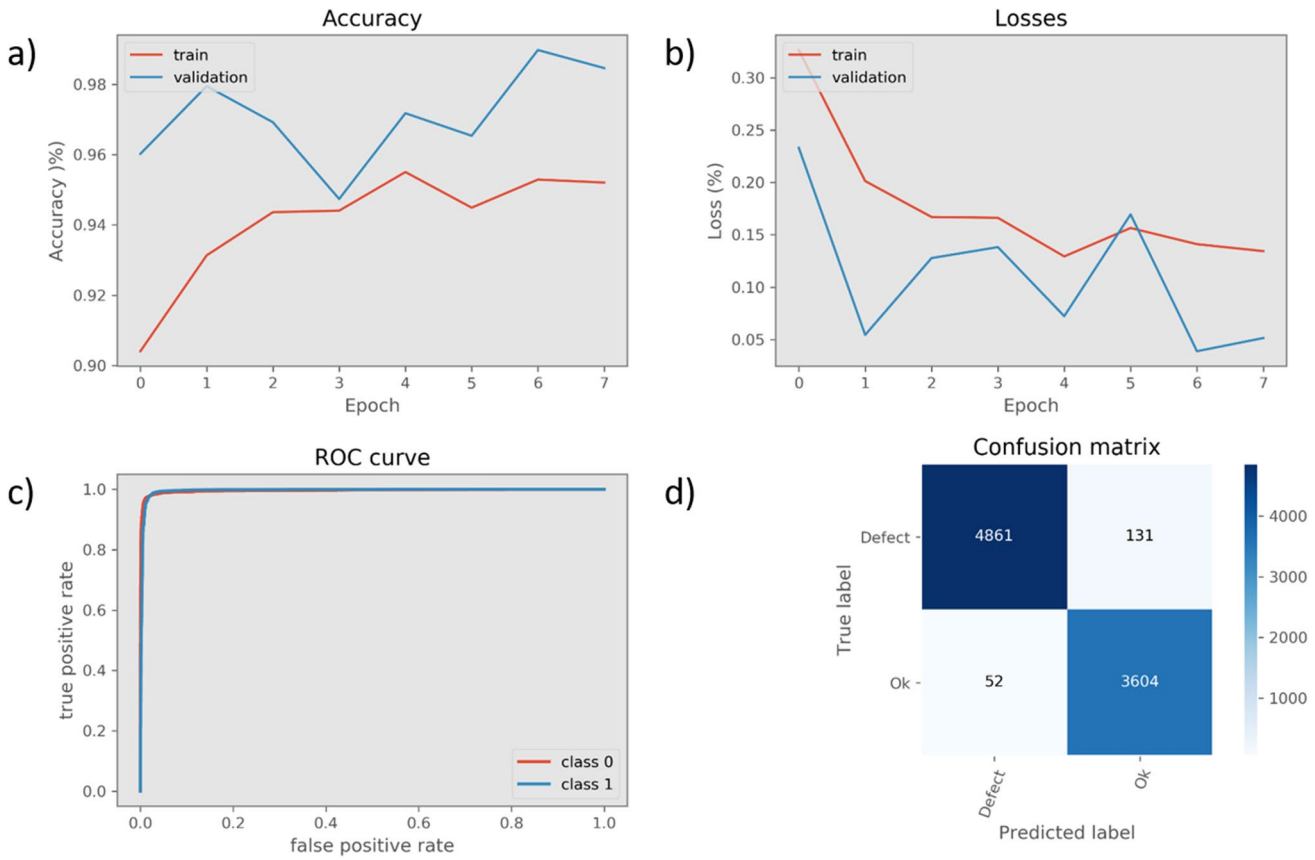

Fig. 2 Results of the Casting dataset. In **a** the training and validation accuracy over the epochs of training are presented. In **b** the losses. In **c** the ROC curves for the two classes, and in **d** the confusion matrix

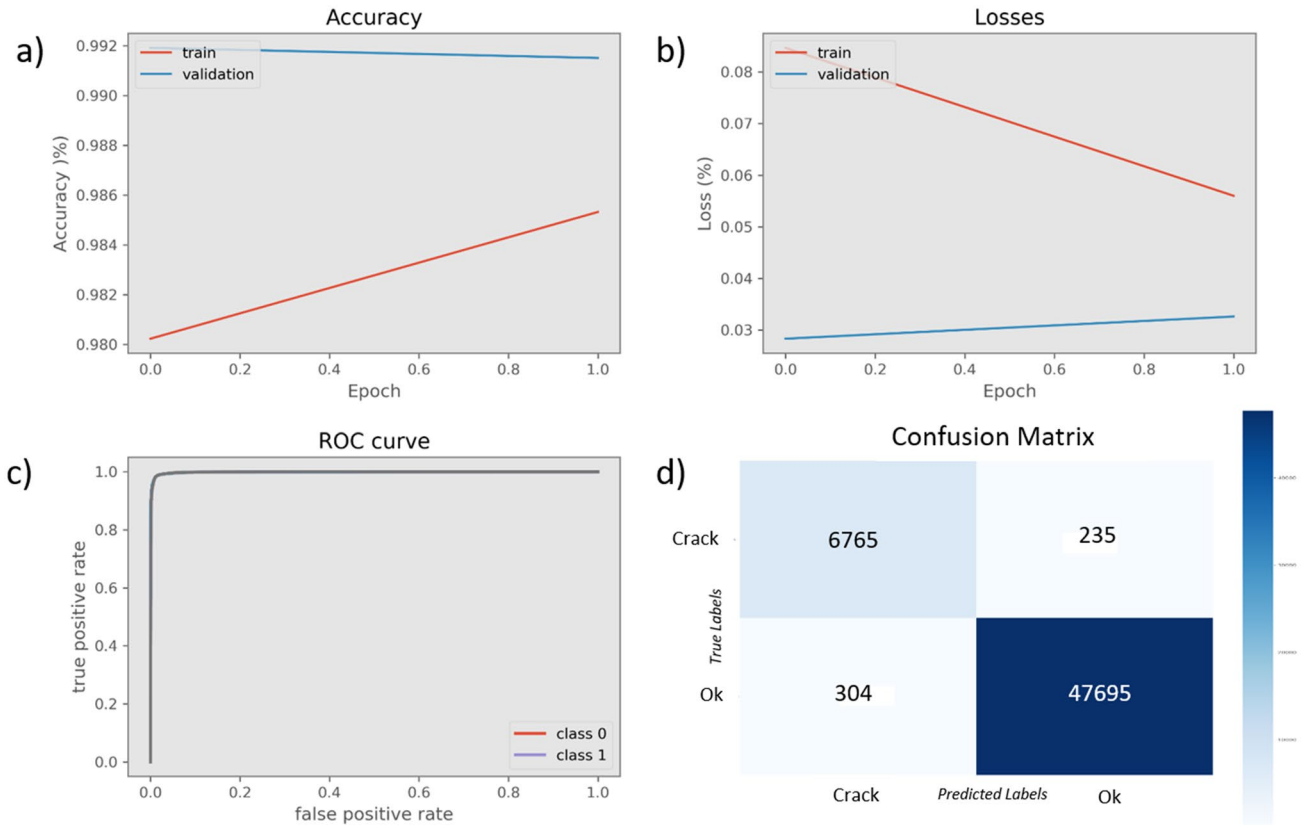

Fig. 3 Results of the Bridge dataset. In **a** the training and validation accuracy over the epochs of training are presented. In **b** the losses. In **c** the ROC curves for the two classes, and in **d** the confusion matrix

Table 4 VGG19 results

Dataset	Accuracy (%)	Precision (%)	Recall (%)	F-1 (%)	AUC (%)
Casting	87.39	78.01	97.72	87.37	97.88
Defect	70.9	66.79	–	–	92.65
Magnetic tile	77.32	70.52	–	–	89.16
Tech	88.29	95.31	–	–	99.6
Bridge	98.72	99.24	99.29	99.26	99.55
Solar (ELPV)	73.85	74.76	58.15	64.97	80.18

Recall and F-1 score was not recorded for the Defect, Magnetic, and Tech datasets, due to the existence of several classes. Those cases are marked with the "–" symbol in the table

F-1 F1 score; *AUC* Area Under Curve score

data (validation set). In epochs 6 and 7, this problem seems to be solved by the model.

Top-performing scores were observed for most of the datasets, while the confusion matrixes confirm the optimal performance of MVGG19 in industrial object and defect recognition challenges.

4.2 Baseline VGG19

The same experiments were performed utilizing the baseline VGG19 sequential structure, to evaluate the proposed multipath methodology, by retaining the parameters and the hyper parameters as described (Table 4).

The results highlight the superiority of MVGG19 over the baseline VGG19 approach in every object and defect recognition dataset utilized in the present study. The comparison between the two networks in terms of the overall accuracy is provided in Fig. 4.

4.3 Other state-of-the-art CNNs

Baseline state-of-the-art Convolutional Neural Networks have been benchmarked to compare the results of MVGG19

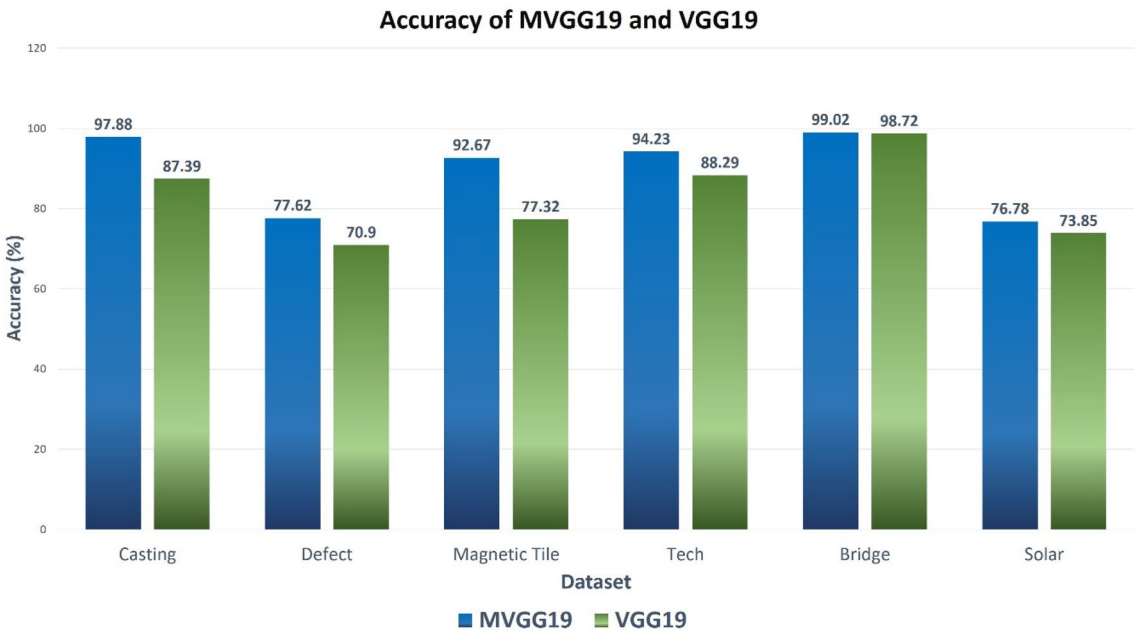

Fig. 4 Comparison of the obtained overall accuracy of MVGG19 and VGG19. The comparison in held in term of the model's accuracy

with the current top performers in similar tasks. All networks were implemented with the following strategy:

a) Borrow the architecture of the CNN
b) Assign every parameter of the CNN as trainable, i.e., do not keep any pre-existing learned parameter from their source-data training.
c) Remove the densely connected layer at the top, which is the baseline classifier and contained pre-learned weights.
d) Place a new densely connected network, the same as MVGG19
e) Use the default optimization parameters to ensure fair comparisons

The results are presented in Table 5. Accuracy and AUC score for state-of-the-art baseline CNNs.

It is observed that MVGG19 comes first in four of the six datasets, whilst coming second in the rest. The reader can observe the results in Fig. 5. It is also observed that neither CNN is capable of achieving better accuracy in Solar and Defect datasets. Every CNN is performing sub-optimally in classifying those two datasets. For the Defect dataset, one reason behind this phenomenon may be the combination of two factors. First, that the database contains many classes, but few images for each class. Second, that the differences between the images are indistinguishable even from the models. This means that more research is needed to achieve greater accuracy, perhaps developing even more sophisticated models in the future. Still, the pleasing AUC score is a promising sign that the models are in the right track in distinguishing between some of the classes. The Solar dataset is more challenging as the low accuracy and AUC scores indicate. The authors intend to perform further research and examination to validate both MVGG19's efficiency and the quality of the dataset itself.

Although the differences in terms of accuracy and AUC are relatively small, one can observe that MVGG19 shows the most consistent results, as shown in Fig. 6.

5 Discussion

Object detection is an essential asset in the industry that could drastically transform some day-to-day operations that are time consuming and, at the same time, expensive. Finding a specific object through visual inspection is a basic task that is involved in multiple industrial processes like sorting, inventory management, machining, quality management, packaging, etc. Until recently, the quality control part of the manufacturing cycle continues to be a difficult task due to its reliance on human-level visual understanding and adaptation to constantly changing conditions and products. With Artificial Intelligence, most of these complications can be handled. AI can automatically distinguish good parts from faulty parts on an assembly line with incredible speed, allowing enough time to take corrective actions. This is a very useful solution for dynamic environments where product environments are constantly changing and time is valuable to the business. Another aspect for further research is manual sorting which involves high cost of labor and accompanying human errors. Even with robots, the process

Table 5 Accuracy and AUC score for state-of-the-art baseline CNNs

Network	Dataset	Accuracy (%)	AUC (%)
Exception network (Chollet 2017)	Casting	97.87	94.24
	Defect	73.03	92.51
	Magnetic	90.5	94.51
	Tech	91.69	96.37
	Bridge	96.84	98.25
	Solar	72.18	79.54
Residual network v.152 (He et al. 2016)	Casting	92.06	94.24
	Defect	70.38	91.32
	Magnetic	87.61	88.37
	Tech	87.99	92.43
	Bridge	98.19	99.42
	Solar	66.08	69.21
Inception network v.3 (Szegedy et al. 2016)	Casting	97.77	98.53
	Defect	72.94	92.48
	Magnetic	91.55	94.26
	Tech	93.35	97.84
	Bridge	94.45	96.83
	Solar	72.48	78.43
Mobile network v.2 (Sandler et al. 2018)	Casting	86.9	92.55
	Defect	66.91	84.32
	Magnetic	**93.48**	96.41
	Tech	91.86	96.55
	Bridge	97.39	98.61
	Solar	74.88	79.26
Dense network v.169 (Huang et al. 2017)	Casting	84.9	89.64
	Defect	69.90	88.37
	Magnetic	88.65	90.35
	Tech	89.52	93.42
	Bridge	94.61	96.54
	Solar	73.81	79.85
Efficient Network v.B0 (Tan and Le 2019)	Casting	88.33	95.31
	Defect	75.28	94.21
	Magnetic	91.79	92.04
	Tech	89.48	94.37
	Bridge	**99.66**	99.05
	Solar	76.1	83.47
Multipath VGG19 (current study)	Casting	**97.88**	**99.59**
	Defect	**77.62**	**94.94**
	Magnetic	92.67	**97.61**
	Tech	**94.23**	**99.94**
	Bridge	99.02	**99.83**
	Solar	**76.78**	**83.36**

Bold values indicate the highest observed score for each dataset

is not accurate enough and is still prone to a discrepancy. AI-powered Object Detection can help transform this tedious and manual process into an efficient and automated process while maintaining the same if not better level of accuracy.

DL networks are a vital part of the algorithms involved in such recognition tasks. For a successful, trustworthy, and global framework to be employed, special-designed models and algorithms are necessary. It is undeniable that not every model is effective in any task. Therefore, the construction of specialized models for industrial image recognition tasks is an interesting and challenging aspect.

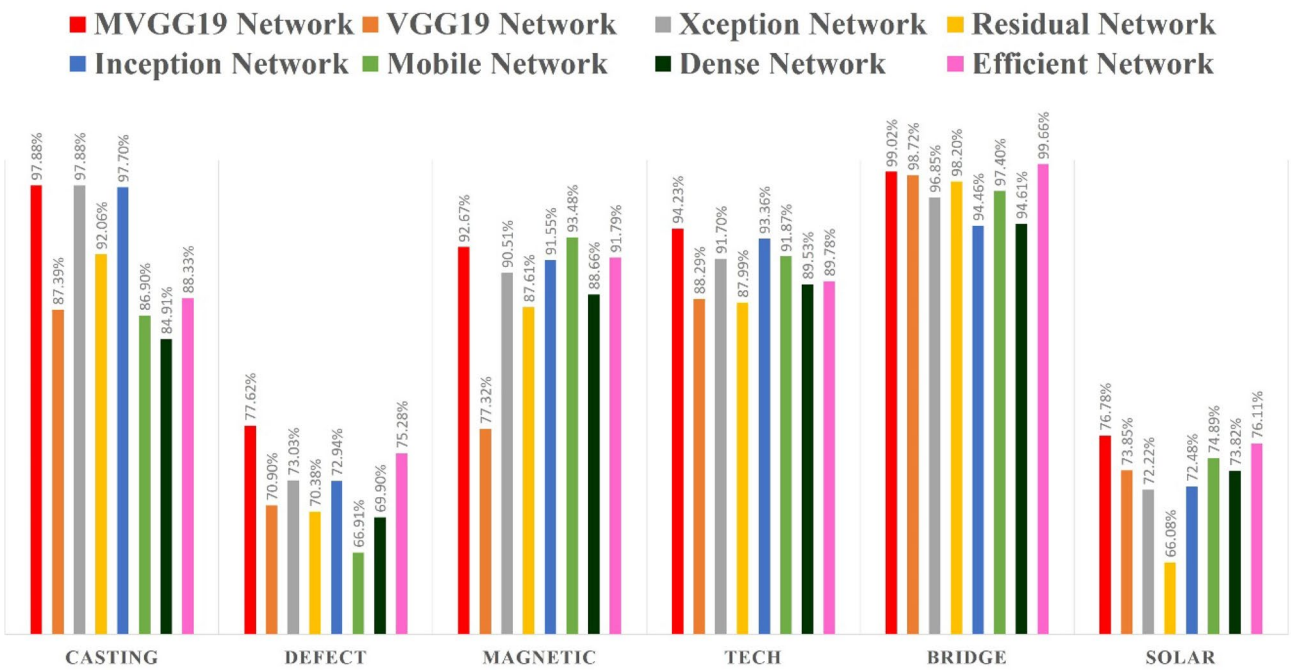

Fig. 5 Accuracy in classification of each dataset by the CNNs of the study

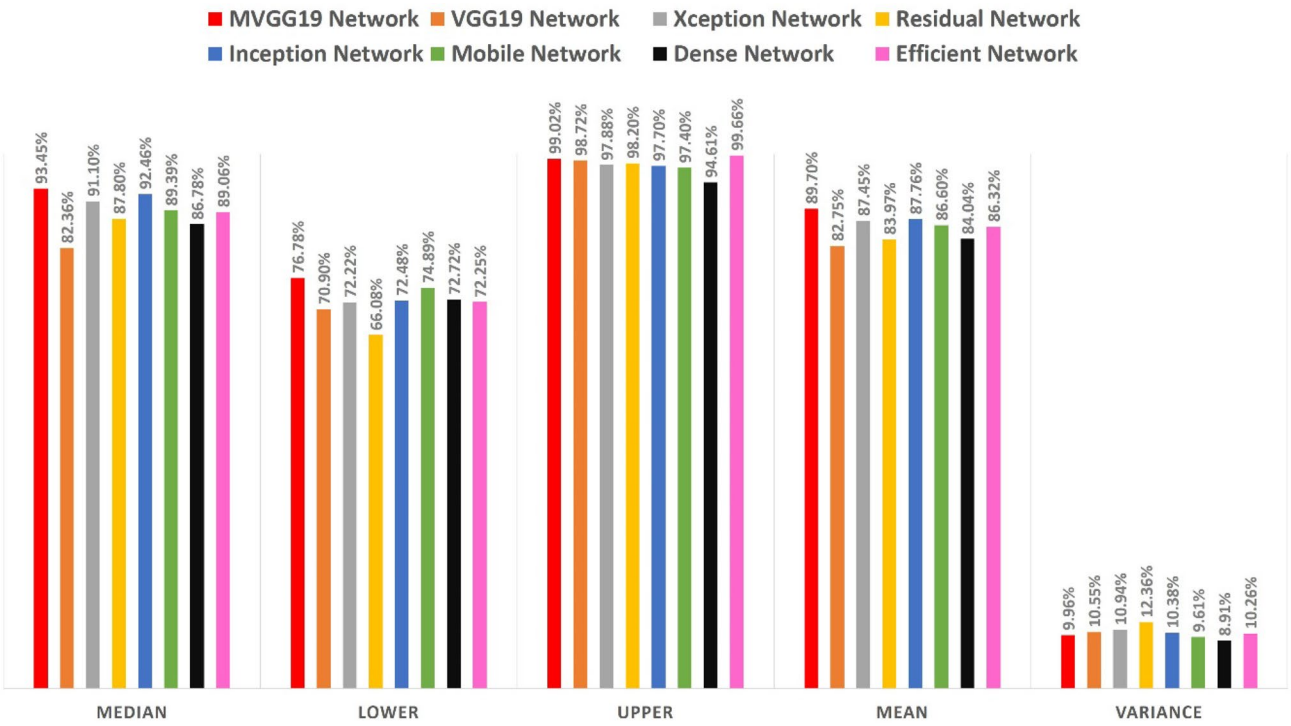

Fig. 6 Median, mean, lower, and upper value, accompanied by the variance of each model for the six datasets of the study

This work paves a way to the development of a generalized DL model capable of recognizing several industry-related objects and defects. Future research and assessment can further verify our results.

6 Conclusions

MVGG19 was evaluated utilizing six different image datasets, consisting of various classes. Its superiority over the baseline VGG19 architecture was demonstrated in five of the six datasets. VGG19 performed quite well, achieving 87.39% accuracy on Casting dataset, 70.9% on the Defect dataset, 77.32% on the Magnetic dataset, 88.29% on the Tech dataset, 98.72% on the Bridge dataset, and 73.85% on the Solar dataset. With MVGG19, the accuracy on the Casting dataset was decreased by 9.77%. The accuracy regarding the Defect dataset was drastically increased by 26.88%. As far as the Magnetic dataset is considered, an improvement of 15.35% was observed. For the Tech dataset, MVGG19 improved the classification accuracy by 5.94%. For the Bridge and the Solar datasets, the accuracy was increased by 0.3% and 2.93%, respectively.

MVGG19 was tested against state-of-the-art baseline CNNs, which were designed for image classification tasks. Although every network was successful in most cases, MVGG19 was found to be the most consistent network. This highlights its appropriateness and specialty in similar tasks, where images come from manufacturing or similar human activities.

The results highlight the effectiveness of the selected feature extraction pipeline. In contrast with the VGG19 architecture, wherein the convolutional layers are stacked in a hierarchical way, MVGG19 allows for staged feature extraction. In this way, feature maps produced by all the convolutional blocks are treated as unique features and not only as derivative features that are simply used by deeper layers to reproduce new ones. The hypothesis that each produced feature map can be significant by itself is confirmed by the results, as MVGG19 succeeds in most of the experimental datasets. In the case of the particular image family, it is concluded that the important features may indeed come from the early layers of the network, which would mean that they are simple low-level features, but decisive ones. This observation implies that the effectiveness of the multipath approach may have extensive application not only to industrial-focused activities but in any activity involving images that can be classified using both simple characteristics (features) and high-level ones.

The contributions of this study are twofold. Firstly, an innovative modification proposal of the successful VGG19 network, called Multipath VGG19 (MVGG19), is proposed and evaluated for defect object and industrial object recognition tasks. The proposed MVGG19 makes full use of each convolution layer and allows for feature fusion, by concatenating the output features of both top and bottom convolutional layers of the network. In this way, the classification is based on an increased variety of features and results in more precise image analysis. This assumption is confirmed by the results of the experiments. Secondly, since the proposed architecture demonstrated its effectiveness in various industrial image datasets, it can be concluded that this architecture can serve as a baseline industrial image classification model, extending its applications in broader areas of manufacturing and transport. Further examination, involving industrial object recognition image datasets is mandatory to confirm and validate the effectiveness of both the multipath feature extraction strategy and the specifically designed MVGG19.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s12652-021-03688-7.

Funding This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Data availability The datasets generated during and/or analysed during the current study are available in the following repositories: Kaggle: Dabhi (2020). Github: https://github.com/lvxiaoming2019/GC10-DET-Metallic-Surface-Defect-Datasets. Github: https://github.com/abin24/Magnetic-tile-defect-datasets. MVTEC: https://www.mvtec.com/company/research/datasets/mvtec-itodd. UTAH State University Libraries: https://digitalcommons.usu.edu/all_datasets/48/. Github: https://github.com/zae-bayern/elpv-dataset

Declarations

Conflict of interest The authors declare that there are no conflicts of interest. The authors have no relevant financial or non-financial interests to disclose.

References

Apostolopoulos ID, Papathanasiou ND, Panayiotakis GS (2021) Classification of lung nodule malignancy in computed tomography imaging utilising generative adversarial networks and semi-supervised transfer learning. Biocybern Biomed Eng 41(4):1243–1257. https://doi.org/10.1016/j.bbe.2021.08.006

Buerhop-Lutz C, Deitsch S, Maier A et al (2018) A benchmark for visual identification of defective solar cells in electroluminescence imagery. In: 35th European photovoltaic solar energy conference and exhibition; pp 1287–1289, 9071 kb. https://doi.org/10.4229/35THEUPVSEC20182018-5CV.3.15

Caggiano A, Zhang J, Alfieri V et al (2019) Machine learning-based image processing for on-line defect recognition in additive manufacturing. CIRP Ann 68:451–454. https://doi.org/10.1016/j.cirp.2019.03.021

Chollet F (2017) Xception: deep learning with depthwise separable convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp 1251–1258. https://doi.org/10.1109/CVPR.2017.195

Dabhi R (2020) Casting product image data for quality inspection, Kaggle Data, v2. https://www.kaggle.com/ravirajsinh45/real-life-industrial-dataset-of-casting-product/metadata. Accessed 1 Sept 2020

Deitsch S, Christlein V, Berger S et al (2019) Automatic classification of defective photovoltaic module cells in electroluminescence images. Sol Energy 185:455–468. https://doi.org/10.1016/j.solener.2019.02.067

Deitsch S, Buerhop-Lutz C, Sovetkin E et al (2020) Segmentation of photovoltaic module cells in electroluminescence images. arXiv:180606530 [cs]

Diez-Olivan A, Del Ser J, Galar D, Sierra B (2019) Data fusion and machine learning for industrial prognosis: trends and perspectives towards Industry 4.0. Inf Fusion 50:92–111. https://doi.org/10.1016/j.inffus.2018.10.005

Drost B, Ulrich M, Bergmann P et al (2017) Introducing mvtec itodd-a dataset for 3d object recognition in industry. In: Proceedings of the IEEE international conference on computer vision workshops. pp 2200–2208. https://doi.org/10.1109/ICCVW.2017.257

Du C, Wang Y, Wang C et al (2020) Selective feature connection mechanism: Concatenating multi-layer CNN features with a feature selector. Pattern Recogn Lett 129:108–114. https://doi.org/10.1016/j.patrec.2019.11.015

Fawzi A, Samulowitz H, Turaga D, Frossard P (2016) Adaptive data augmentation for image classification. In: 2016 IEEE international conference on image processing (ICIP). IEEE, pp 3688–3692. https://doi.org/10.1109/ICIP.2016.7533048

Fu G, Sun P, Zhu W et al (2019) A deep-learning-based approach for fast and robust steel surface defects classification. Opt Lasers Eng 121:397–405. https://doi.org/10.1016/j.optlaseng.2019.05.005

Goodfellow I, Bengio Y, Courville A, Bengio Y (2016) Deep learning. MIT press, Cambridge

Gu J, Wang Z, Kuen J et al (2017) Recent advances in convolutional neural networks. arXiv:151207108 [cs]

Han J, Zhang D, Cheng G et al (2018) Advanced deep-learning techniques for salient and category-specific object detection: a survey. IEEE Signal Process Mag 35:84–100. https://doi.org/10.1109/MSP.2017.2749125

He K, Zhang X, Ren S, Sun J (2016) Identity mappings in deep residual networks. European conference on computer vision. Springer, New York, pp 630–645. https://doi.org/10.1007/978-3-319-46493-0_38

Huang G, Liu Z, Van Der Maaten L, Weinberger KQ (2017) Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp 4700–4708

Huang Y, Qiu C, Guo Y et al (2018) Surface defect saliency of magnetic tile. In: 2018 IEEE 14th international conference on automation science and engineering (CASE). IEEE, Munich, pp 612–617. https://doi.org/10.1007/s00371-018-1588-5

Iandola FN, Han S, Moskewicz MW, et al (2016) SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size. arXiv preprint arXiv:160207360

Khan S, Rahmani H, Shah SAA, Bennamoun M (2018) A guide to convolutional neural networks for computer vision. Synth Lect Comput vis 8:1–207. https://doi.org/10.2200/S00822ED1V01Y201712COV015

Khan S, Yong S-P (2016) A comparison of deep learning and hand crafted features in medical image modality classification. In: 2016 3rd international conference on computer and information sciences (ICCOINS). IEEE, pp 633–638. https://doi.org/10.1109/ICCOINS.2016.7783289

Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. arXiv preprint arXiv:14126980

Kornblith S, Shlens J, Le QV (2019) Do better imagenet models transfer better? In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp 2661–2671. https://doi.org/10.1109/CVPR.2019.00277

Krizhevsky A, Sutskever I, Hinton GE (2017) ImageNet classification with deep convolutional neural networks. Commun ACM 60:84–90. https://doi.org/10.1145/3065386

LeCun Y, Bengio Y, Hinton G (2015) Deep learning. Nature 521:436–444. https://doi.org/10.1038/nature14539

Lin Y, Nie Z, Ma H (2017) Structural damage detection with automatic feature-extraction through deep learning. Comput Aided Civ Infrastruct Eng 32:1025–1046. https://doi.org/10.1111/mice.12313

Lv X, Duan F, Jiang J et al (2020) Deep metallic surface defect detection: the new benchmark and detection network. Sensors 20:1562. https://doi.org/10.3390/s20061562

Maguire M, Dorafshan S, Thomas RJ (2018) SDNET2018: a concrete crack image dataset for machine learning applications. https://doi.org/10.15142/T3TD19

Mehta S, Paunwala C, Vaidya B (2019) CNN based traffic sign classification using adam optimizer. In: 2019 international conference on intelligent computing and control systems (ICCS). IEEE, pp 1293–1298. https://doi.org/10.1109/ICCS45141.2019.9065537

Najafabadi MM, Villanustre F, Khoshgoftaar TM et al (2015) Deep learning applications and challenges in big data analytics. J Big Data 2:1–21. https://doi.org/10.1186/s40537-014-0007-7

Poernomo A, Kang D-K (2018) Biased dropout and crossmap dropout: learning towards effective dropout regularization in convolutional neural network. Neural Netw 104:60–67. https://doi.org/10.1016/j.neunet.2018.03.016

Sandler M, Howard A, Zhu M et al (2018) Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp 4510–4520

Shin H-C, Roth HR, Gao M et al (2016) Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. IEEE Trans Med Imaging 35:1285–1298. https://doi.org/10.1109/TMI.2016.2528162

Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:14091556

Szegedy C, Vanhoucke V, Ioffe S et al (2016) Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp 2818–2826. https://doi.org/10.1109/CVPR.2016.308

Tajbakhsh N, Shin JY, Gurudu SR et al (2016) Convolutional neural networks for medical image analysis: full training or fine tuning? IEEE Trans Med Imaging 35:1299–1312. https://doi.org/10.1109/TMI.2016.2535302

Tan M, Le Q (2019) Efficientnet: rethinking model scaling for convolutional neural networks. In: International conference on machine learning. PMLR, pp 6105–6114

Wagner J, Schiller D, Seiderer A, André E (2018) Deep learning in paralinguistic recognition tasks: are hand-crafted features still relevant?. https://doi.org/10.21437/Interspeech.2018-1238

Wang J, Ma Y, Zhang L et al (2018a) Deep learning for smart manufacturing: methods and applications. J Manuf Syst 48:144–156. https://doi.org/10.1016/j.jmsy.2018.01.003

Wang P, Liu H, Wang L, Gao RX (2018b) Deep learning-based human motion recognition for predictive context-aware human-robot collaboration. CIRP Ann 67:17–20. https://doi.org/10.1016/j.cirp.2018.04.066

Wang J, Fu P, Gao RX (2019) Machine vision intelligence for product defect inspection based on deep learning and Hough transform. J Manuf Syst 51:52–60. https://doi.org/10.1016/j.jmsy.2019.03.002

Weiss K, Khoshgoftaar TM, Wang D (2016) A survey of transfer learning. J Big Data 3:9. https://doi.org/10.1186/s40537-016-0043-6

Wong SC, Gatt A, Stamatescu V, McDonnell MD (2016) Understanding data augmentation for classification: when to warp? In: 2016 international conference on digital image computing: techniques

and applications (DICTA). IEEE, pp 1–6. https://doi.org/10.1109/DICTA.2016.7797091

Yan LC, Yoshua B, Geoffrey H (2015) Deep learning. Nature 521:436–444. https://doi.org/10.1038/nature14539

Yue-Hei Ng J, Yang F, Davis LS (2015) Exploiting local features from deep networks for image retrieval. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp 53–61. https://doi.org/10.1109/CVPRW.2015.7301272

Zeng G, Zhou J, Jia X et al (2018) Hand-crafted feature guided deep learning for facial expression recognition. In: 2018 13th IEEE international conference on automatic face & gesture recognition (FG 2018). IEEE, pp 423–430. https://doi.org/10.1109/FG.2018.00068

Zhuang F, Qi Z, Duan K et al (2020) A comprehensive survey on transfer learning. Proc IEEE 109:43–76. https://doi.org/10.1109/JPROC.2020.3004555

Zoph B, Cubuk ED, Ghiasi G et al (2020) Learning data augmentation strategies for object detection. European conference on computer vision. Springer, New York, pp 566–583. https://doi.org/10.1007/978-3-030-58583-9_34