Waveguide quality inspection in quantum cascade lasers: A capsule neural network approach

Abd Al Rahman M. Abu Ebayyeh [a,b,*], Alireza Mousavi [b], Sebelan Danishvar [b], Stéphane Blaser [c], Tobias Gresch [c], Olivier Landry [c], Antoine Müller [c]

[a] *Department of Electrical and Electronic Engineering, Imperial College London, London SW7 2AZ, UK*
[b] *Department of Electronic and Computer Engineering, Brunel University London, Uxbridge UB8 3PH, UK*
[c] *Alpes Lasers SA, Avenue des Pâquiers 1, CH-2072 Saint-Blaise, Switzerland*

A R T I C L E   I N F O

A B S T R A C T

Growing demand for consumer electronic devices and telecommunications is expected to drive the quantum cascade laser (QCL) market. The increase in the production rate of QCLs increases the likelihood of production failures and anomalies. The detection of waveguide defects and dirt using automatic optical inspection (AOI) and deep learning (DL) is the main focus of this study. The images samples of QCLs were collected from a laser manufacturing plant in Europe. Due to the lack of sufficient dirt and defect samples, automatic and manual data augmentation approaches were used to increase the number of images. A combination of an improved capsule neural network (WaferCaps) and convolutional neural network (CNN) based on parallel decision fusion is used to classify the samples. The output of these classifiers were combined based on rule-based selection algorithm that chooses the performance of the best classifier according to the class. The proposed approach was compared with the performance of standalone models, different state-of-the-art DL models such as CapsNet, ResNet-50, MobileNet, DenseNet, Xception and Inception-V3 and other machine learning (ML) models such as Support Vector Machine (SVM), decision tree, k-NN and Multi-layer Perceptron (MLP). The proposed approach outperformed them all with a validation accuracy of 98.5%.

1. Introduction

High-performance indium phosphide (InP) quantum cascade lasers (QCLs) play a key role in many optoelectronic applications in the mid to far infrared (IR) regions such as IR imaging and spectroscopy (Razeghi, 2009). QCLs are unipolar semiconductor lasers that emit in the range between 4 and 12 μm of wavelength. The first QCL was invented and experimented in 1994 at Bell labs by Faist et al. (1994). Since then and due to the extensive knowledge acquired in the field of epi-material growth, wafer processing, and die packaging, QCLs were developed rapidly (Curl et al., 2010; Figueiredo et al., 2017). Beside their unique combination of compactness, room temperature operation, high power output and narrow linewidth, QCLs are well-known in their tuning abilities of IR light (Villa et al., 2019). A typical QCL wafer (Fig. 1) consists of multiple laser devices and is made of a stack of InGaAs/AlInAs layers on InP substrate using a series of chemical processes such as

molecular beam epitaxy (MBE) or metal–organic chemical vapor deposition (MOCVD) (Pecharroman-Gallego, 2017). Like other semiconductor lasers, each QCL in the wafer consists of a laser core, a waveguide (Fig. 2), and a feedback mechanism (Razeghi, 2009).

Due to the complexity of the fabrication process, QCL wafers are prone to multiple flaws. Where some of these malfunctions can only be detected using special tests, a discontinuity in the waveguide of the QCL can be detected visually with the aid of microscope and digital camera. This defect occurs when the hard-mask defining the waveguide is locally damaged and detaches during the etching process of the waveguide. The detachment causes the waveguide to be damaged or interrupted. Another common problem is the settlement of dirt over the waveguide of the QCL. Although dirt does not necessarily cause damage to the waveguide, but the inspector should be alerted to this case since a damage can happen beneath the dirt on the waveguide. Dirt inspection on the waveguide is visually conducted using microscope and digital camera.

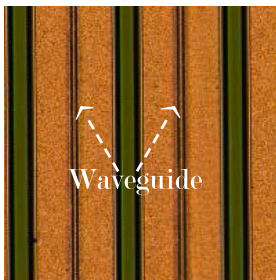**Fig. 1.** Optoelectronic wafer that contains multiple QCL devices.

Fig. 2. A zoomed-in sample that illustrates the waveguide of QCL in optoelectronic wafer.

High-power QCL have cavity lengths between 6 and 9 mm and the waveguide needs to be in impeccable condition to ensure reliable operation. Therefore, quality monitoring of the waveguide is essential.

Defects and dirt detection on the waveguide in majority of cases are conducted manually in small and make to order (low volume high variety) plants. In order to increase productivity and production rate to meet demand, automatic optical inspection (AOI) techniques can be utilized to automate and increase the accuracy of the inspection and quality control task (Abu Ebayyeh & Mousavi, 2020; Wang & Huang, 2004). Furthermore, AOI would reduce the rate of false alarms caused by color perception induced human error (Kim et al., 2009). It is therefore safe to claim that a bespoke robust high quality AOI system could be a desirable tool for detection of anomalies for QCL wafers (Abu Ebayyeh & Mousavi, 2020; Edinbarough et al., 2005).

Conventional machine learning (ML) algorithms including supervised and unsupervised have been widely used in many AOI tasks for identifying anomalies in different industries such as food (Portalés & Ribes-Gómez, 2015), tiles (Iglesias et al., 2018), metals (Cruz et al., 2021), medical analysis (Muzammal et al., 2020) and electronics (Abu Ebayyeh & Mousavi, 2020; Huang & Pan, 2015). Despite its powerful performance, conventional ML methods are only limited to process structured data (e.g. images and natural language) in its raw original form (Westphal & Seitz, 2021). The rise of computational power, big data and graphical processing units (GPUs) brought the application of deep learning (DL) techniques to closer to solving actual and real manufacturing process applications and replace conventional ML methods (Danishvar et al., 2021; Sodhro et al., 2019). DL networks allows to process the raw data and perform feature extraction automatically due to the multiple and complex layers that they have. Convolutional neural networks (CNNs), autoencoders and capsule networks (CapsNets) are well-known DL networks that are currently used

for AOI tasks due to their ability in processing complex and colored images (Abu Ebayyeh & Mousavi, 2020; Westphal & Seitz, 2021).

In this study, a bespoke dedicated vision system equipped with a novel DL solution that combines CNN and improved capsule network (WaferCaps) is proposed for classifying the waveguide anomalies. The solution is tested and validated in an actual and real QCL wafers collected from a laser manufacturing plant. In Section 2, the related work in this subject area is presented. In Section 3, the image acquisition and preprocessing methods are discussed. For the purpose of classifying the images into three classes, namely, normal, dirt and defect, a data augmentation approach is proposed for upsampling the dataset in order to utilize them in the training of the DL network. In Sections Section 4, the outcomes of the proposed solution is discussed. The discussions are enriched by comparing our main approach with alternative existing DL models. The conclusions section (Section 5) conclude the article. This work could reduce the burden on human inspectors that usually use microscope to detect waveguide anomalies of QCLs. The framework of the proposed approach can be illustrated in Fig. 3.

2. Related work

Based on our literature review in public domain and limited interviews with industry experts we have not found dedicated research that has specifically dealt with QCL anomalies using AOI techniques; nevertheless, the most relevant research articles that used similar techniques in defect inspection are discussed.

Chien et al. (2020) proposed two CNN models for classifying semiconductor wafer surface defects; region-based CNN (R-CNN) and CNN with five convolutional layers. The dataset used were open source and is called WM-811k, which contains eight defects patterns in binary image format (failed dies are represented with logic 1 and normal dies with logic 0) and called wafer bin maps (WBM). The performance of the two models was very close with a validation of 95%. However, this study did not consider all type of defects such as mixed type defects. Furthermore, since binary images were considered, the real defect pattern may not be accurate enough due to the information loss in such images (Abdiyeva et al., 2020). Liu et al. (2019) proposed DL approach based on U-shaped deep residual neural network (U-ResNet) to detect conductive particles in Thin Film Transistor Liquid Crystal Display (TFT-LCD). The images acquired were in micrometer scale and were divided into 3 datasets using 8-bit grayscale format. Despite of the considerably low computation time ($\leq$4s/product), the proposed algorithm performance was poor on the validation dataset. Furthermore, the lack of high quality images of this study played a vital role in the misclassification rate. Yu et al. (2019) proposed stacked convolutional sparse denoising autoencoder (SCSDAE) for classifying WBM defect patterns. The dataset used was divided into two types; simulated and industrial in which the detection accuracy rate in this study was shown to be 95.13% in the simulation case and 94.75% in the industrial case. However, the problem of limited training data caused SCSDAE to overfit. Furthermore, due to labeled data scarcity for some patterns in WBMs, it is a difficult for SCSDAE to implement WMPR on imbalanced dataset. Mei et al. (2018) proposed multiscale convolutional denoising autoencoder (MSCDAE) model to detect texture surface defects such as LCD panels, ceramic tiles, and textiles. The proposed MSCDAE model applies a three-layer pyramid to perform multiscale defect inspection. However, their approach requires a separate model to be trained and tested for each layer; therefore, it is time-consuming and leads to excessive memory consumption (Abu Ebayyeh & Mousavi, 2020). Furthermore, their approach achieved a reasonable recall but relatively low precision (Yang et al., 2019). Abu Ebayyeh et al. (2022) proposed an improved CapsNet (WaferCaps) for classification of WBM pattern defects. In their approach, they first used deep convolutional neural network (DCGAN) for increasing the dataset and balancing them. Then they used WaferCaps for classifying the original and augmented data. Their proposed approach achieved a relatively high accuracy where
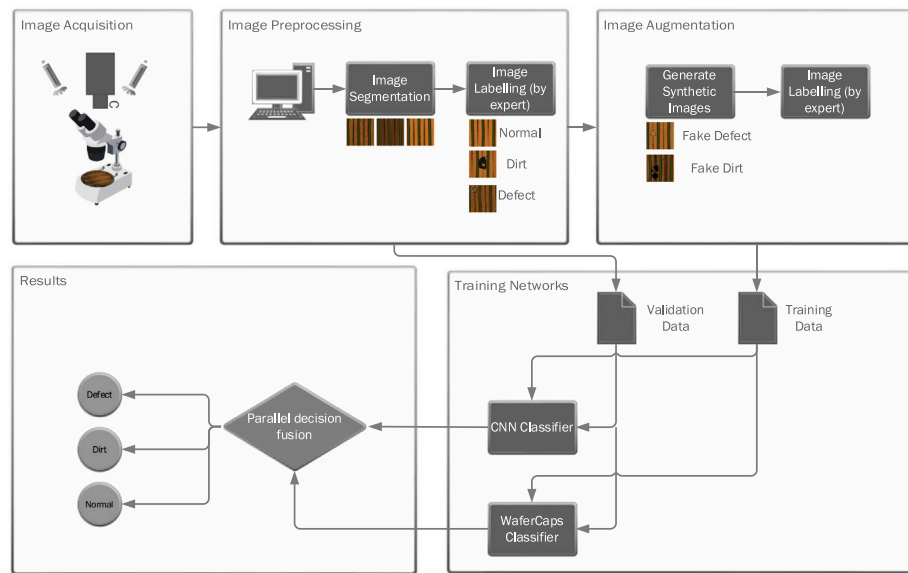
Fig. 3. A detailed framework for the proposed approach in this study.

the training, validation and test accuracies were 99.59%, 97.53%, and 91.4% respectively.

In this research, the WaferCaps from Abu Ebayyeh et al. (2022) were enhanced by combining its performance with a CNN model in a parallel decision fusion approach and choosing the best predictions of both using rule-based selection. Compared with the work conducted in Abu Ebayyeh et al. (2022), the performance of the classifier where much improved. Furthermore, the validation accuracy where enhanced compared with the other previous research reviewed in this section that used semiconductor wafers. Hence, the data of this research has not been used elsewhere and therefore, the comparison is based on the detection accuracy only.

To the best of our knowledge, there was not any research articles that tackled QCL anomalies or any optoelectronic components. Therefore, motivated by the lack of research in the area of identifying anomalies in QCLs, the main objective of this research is to develop an accurate AOI system based on DL that can be used to classify waveguide anomalies (i.e defect and dirt) of QCLs. One of the significant challenges faced on the development of this system is the lack of enough samples to train the DL algorithm. Therefore, manual and automatic data augmentation approaches were proposed to tackle this problem.

This paper's primary contributions are as follows:

- Providing DL framework to assess the quality of the waveguide in QCLs.
- Generating waveguide samples library by applying manual and automatic augmentation approaches of the minority classes.
- Integrate the performance of CNN and an improved capsule network (WaferCaps) using parallel decision fusion to deploy the classification of the anomaly and normal samples.
- Compare the performance of the proposed approach with standalone models (without combination), several state-of-the-art DL algorithms such as CapsNet (Sabour et al., 2017), ResNet50 (He et al., 2016), DenseNet (Huang et al., 2017), MobileNet (Howard et al., 2017), Inception-V3 (Szegedy et al., 2016), and Xception (Chollet, 2017) and other well-known ML algorithms such as support vector machine (SVM), decision tree, k-NN and multi-layer perceptron (MLP).

3. Methodology

The data collected as well as the solution provided are implemented in laser manufacturing plant in Europe where the inspection

process of optoelectronic wafers was applied. The working stations include a cleaving station, a manual die bonder, a facet inspection microscope, and various optical setups. In this section, the process of automating the process is explained and described in terms of the image acquisition procedure, image handling, image augmentation and classification procedures are presented. The main focus is to classify the waveguide of QCL into three classes namely, normal, defect and dirt using DL and computer vision techniques. Fig. 3 demonstrate the complete framework applied in this study.

3.1. Investigated anomalies

Two commonly occurring types of anomalies in QCL wafer manufacturing are considered, defected waveguide and dirt. Defected waveguides can be recognized visually by the discontinuity of the black lines that represent the waveguide as in Fig. 4. Dirt anomaly occurs when the waveguide is covered with external material that are represented in the form of black clusters (Fig. 5).

Fig. 6 shows normal (non-defective) samples of waveguide. The DL algorithm used for detection will classify the sample images according to three classes; Defect, Normal and Dirt.

3.2. Image acquisition and augmentation

Images of the opto-electronic wafer are acquired at premises using a S-Neox optical profilometer, by Sensofar, using a microscope and digital camera objective with 5x magnification and white light illumination. For coverage of a full 2″ wafer, about 400 individual images are acquired in a raster-scan, while adjusting for the image focus to compensate for wafer bending. The individual images are subsequently stitched together into a single image, normalized to $20,000 \times 20,000$ pixels and manually corrected for rotation to obtain vertical waveguide using sensofar tool that has a controlled x–y table. This step is important in analyzing the waveguides while maintained in vertical position and is illustrated in Figs. 1 and 7(d) shows the complete wafer image acquired.

Next step is automatic segmentation process. The images are broken down into multiple segments such that each wafer will result into up to 430 images. In this study, three wafer images are segmented. These segmented images are then labeled according to expert view as defect, dirt or normal. The observed samples during the experiment contained 1,000 of normal, 36 defective and 240 dirt classes.
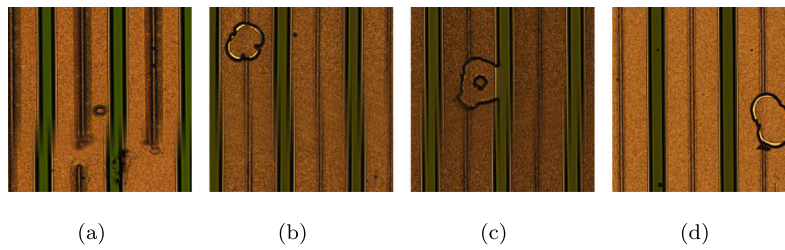
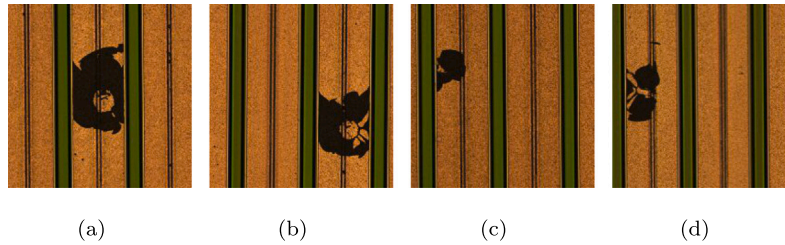**Fig. 4.** Samples for waveguide defects in QCL wafer.

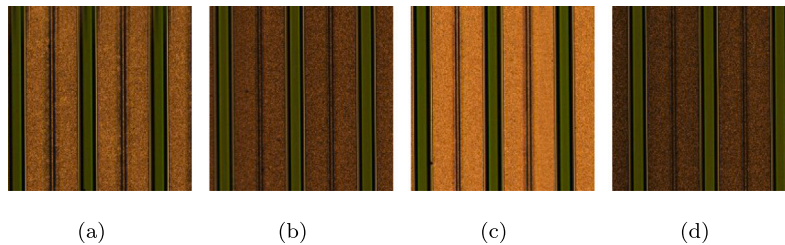**Fig. 5.** Samples for dirt on waveguide in QCL wafer.

Fig. 6. Normal waveguide samples in QCL wafer.

One of the main challenges of the deploying learning models is limitation of available data (analysis span) and lack of sufficient historical image library, therefore risk of overfitting and low validation accuracy. In our case, this challenge is visible in the defect and dirt samples. One way to mitigate the situation would be image augmentation. Augmentation involves creating new dataset based on existing data for the purpose of increasing the number of samples. A combination of automatic and manual augmenting was implemented to increase the number of dirt and defect samples to match the normal samples. In the manual augmentation synthetic defect and dirt features were transposed on the images (Fig. 8). The large library of produced synthetic images were validated and verified by experts on whether the could be realistic occurrences.

The automatic augmentation approach applies different image transformation operations on the original and manually augmented images such as cropping, flipping, scaling, mirroring and scaling to further increase the dataset for training.

3.3. Networks architecture

In this section, the two network architectures used in this study are explained as well as the proposed approach in integrating the performance of CNN and the improved capsule network WaferCaps.

3.3.1. CNN

CNNs have been widely preferred in machine vision tasks due to their high image recognition capabilities (Abu Ebayyeh & Mousavi, 2020). Else than classification, CNNs can perform feature extraction on the target images via the networks' layers and topology. A typical CNN consists of different convolutional, pooling and fully connected (FC) layers. Convolutional and pooling layers distinguish a CNN from

normal MLP, they are also responsible for the feature extraction part in the network. A convolutional layer performs element-by-element multiplication between input matrix (image) and kernel (filter) of different weights. The weights are randomly generated at the beginning of the training. This multiplication will result in a feature map that will be subjected to similar operations. The convolution process is explained in Fig. 9(a).

A pooling layer is usually performed between two convolutional operations to reduce the spatial size of the feature map which also reduces the computational time. Pooling layers are important for selecting the important features in the image and discarding the redundant ones. Max pooling is popular method in pooling, in which the maximum feature value is selected from the feature map according to the kernel size selected as shown in Fig. 9(b). FC layers are responsible in encoding the features from previous layers in order to come up with relevant class and hence perform classification. In this research, the network shown in Fig. 10 was used to combine its performance with WaferCaps as will be found in Section 3.3.3. This network consists of four convolutional layer, four pooling layers and two FC layers. Dropout was used between the final two FC layers. The basic idea behind dropout is to randomly disconnect connections between neurons in connected layers according to specific dropout rate. Therefore, by reducing coadaptations, a network can generalize training examples more efficiently (Cha et al., 2017) (see Table 1).

3.3.2. CapsNet

CapsNet was proposed by Sabour et al. (2017). It was originally used in classifying handwritten digits of the popular MNIST dataset into ten classes (0–9). The architecture of the network has some similarities with CNN in terms of the convolutional layers used. However, layer-based squashing, dynamic routing and lack of pooling layers are three

(a) Raster scan with sensofar tool

(b) Stitching with stitching-tool provided by sensofar

(c) Rotation of the image such that waveguides are oriented vertically

(d) Cropping to $20,000 \times 20,000$ pixels
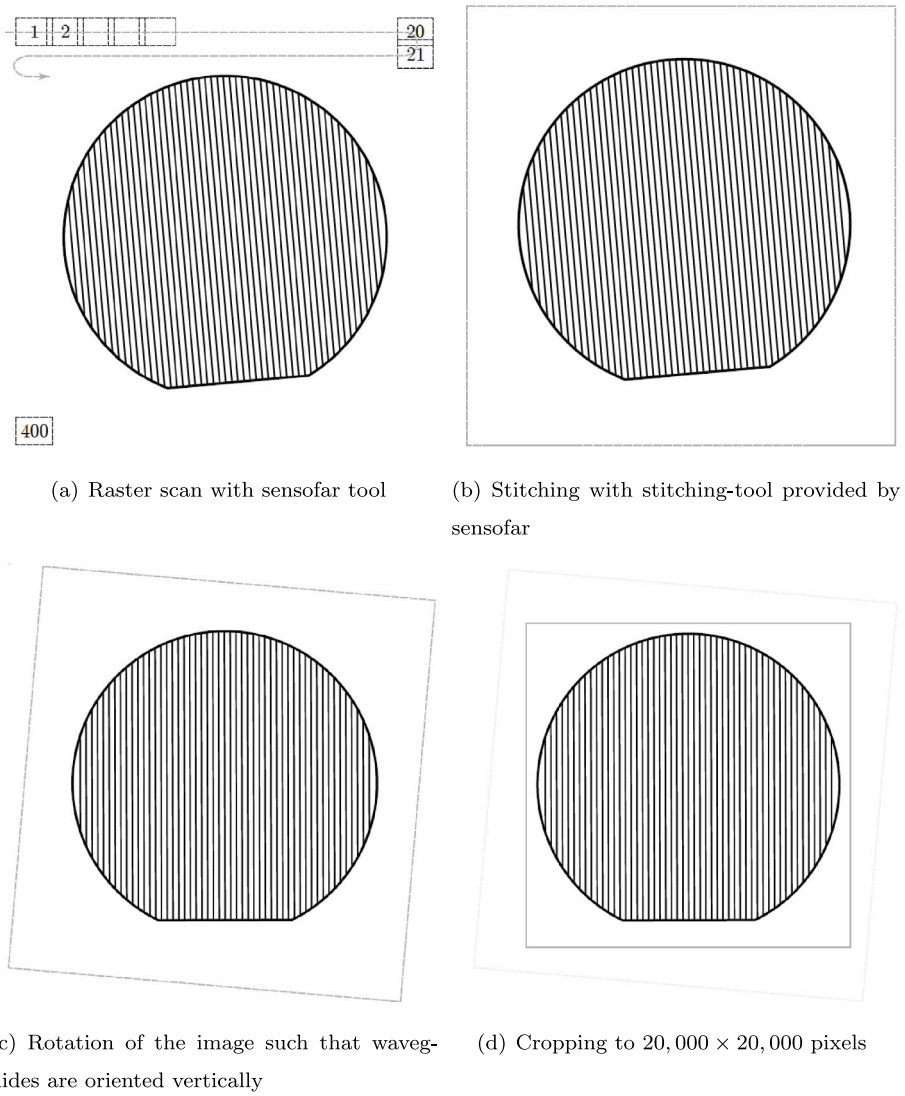
Fig. 7. Image acquisition approach.

(a)　　　　(b)　　　　(c)　　　　(d)

(e)　　　　(f)　　　　(g)　　　　(h)

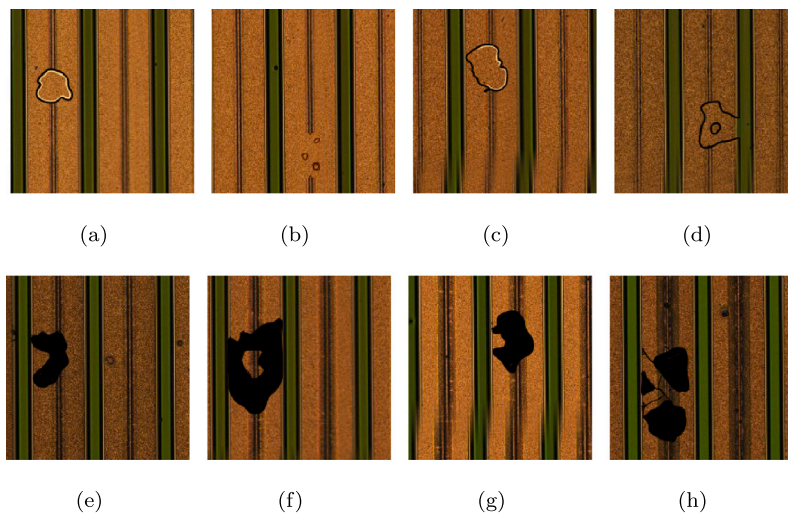**Fig. 8.** Manually augmented samples for defect and dirt anomalies.

(a) Convolutional operation
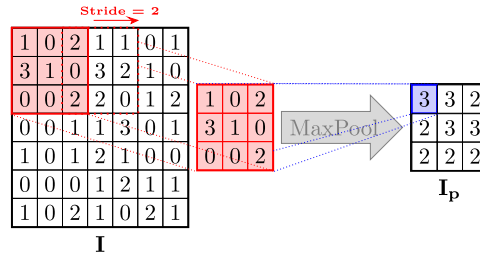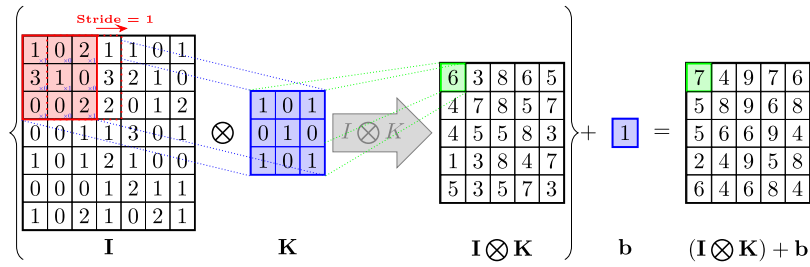

(b) Pooling operation

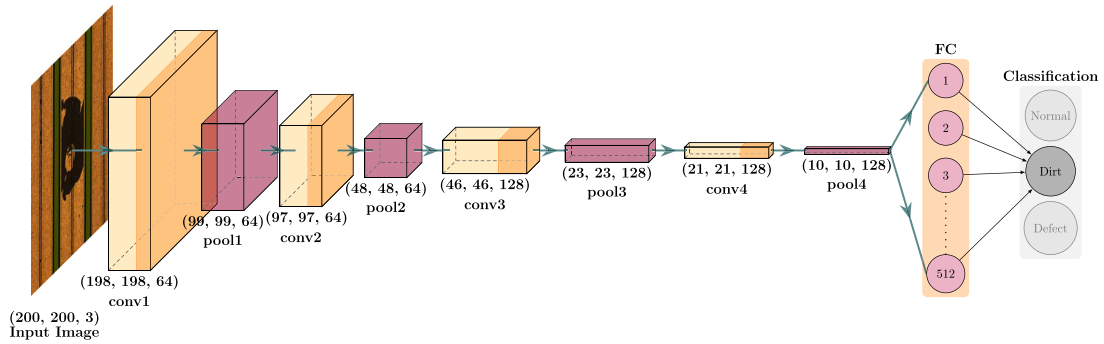**Fig. 9.** Convolution and pooling operations in CNNs.

Fig. 10. A proposed CNN architecture for classification and decision fusion approach.

Table 1
Layers structure and parameters used of the considered CNN for classification and decision fusion approach.

Layer	Type	Input size	Kernel size/stride	Activation	Dropout	Output size
1	conv	[200, 200, 3]	3/1	ReLU	No	[198, 198, 64]
2	pool	[198, 198, 64]	2/1	–	No	[99, 99, 64]
3	conv	[99, 99, 64]	3/1	ReLU	No	[97, 97, 64]
4	pool	[97, 97, 64]	2/1	–	No	[48, 48, 64]
5	conv	[48, 48, 64]	3/1	ReLU	No	[46, 46, 128]
6	pool	[46, 46, 128]	2/1	–	No	[23, 23, 128]
7	conv	[23, 23, 128]	3/1	ReLU	No	[21, 21, 128]
8	pool	[21, 21, 128]	2/1	–	No	[10, 10, 128]
9	FC	[10, 10, 128]	–	ReLU	Yes	[12800,]
10	FC	[12800,]	–	Softmax	No	[3,]

Table 2
Layers of MNIST CapsNet proposed by Sabour et al. (2017).

Layer	Type	Input size	Kernel size/stride	Activation	Dropout	Output size
1	conv1	[28, 28, 1]	9/1	ReLU	No	[20, 20, 265]
2	PrimCaps	[20, 20, 265]	9/2	ReLU	No	[6, 6, 8, 32]
3	DigitCaps	[6, 6, 8, 32]	–	Squash	No	[16, 10]
4	FC	[16, 10]	–	Softmax	No	[10,]

main aspects that distinguish CapsNet from CNN. CapsNet replaces the scalar-output feature detectors of CNNs with vector-output capsules and replaces pooling with routing-by agreement. The lack of pooling layers in CapsNets can be considered an advantage in terms of preserving the features extracted in the image; however it comes with cost of increasing the computational time required. CapsNet replaces the scalar-output feature detectors of CNNs with vector-output capsules and replaces pooling operation with routing-by agreement. Each neuron in the capsule represents various features in particular parts of an image,

such that the spatial location of the feature can be preserved. Fig. 11 and Table 2 represent the original CapsNet architecture as proposed by Sabour et al. (2017).

The first layers in the network is conventional convolutional layers as the one in the CNN which performs convolution operation as described in Fig. 9(a). In the second layer (PrimaryCaps), each capsule i of the 32 has an activity vector $\mathbf{u}_i$ that is used to represent spatial information in the form of instantiation parameters. The output of $\mathbf{u}_i$ is then passed to the next layer (DigitCaps), where each capsule j from the 16 per digit class receives $\mathbf{u}_i$ and multiplies it by the weight matrix $\mathbf{W}_{ij}$. As a result, the prediction vector $\hat{\mathbf{u}}_{j|i}$ will be generated, indicating the amount of contribution for capsule i in the PrimaryCaps on capsule j in the DigitCaps, as given by Eq. (1).

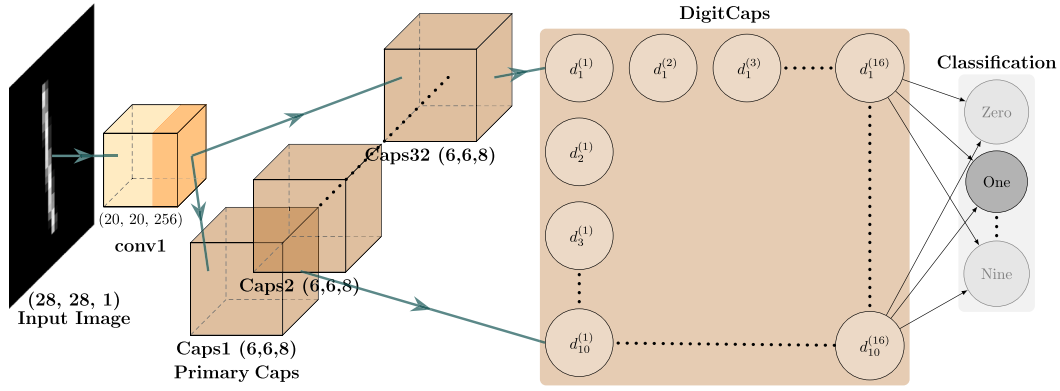$$\hat{\mathbf{u}}_{j|i} = \mathbf{W}_{ij}\mathbf{u}_i \tag{1}$$

Fig. 11. Originally proposed CapsNet for MNIST handwritten digit classification (Sabour et al., 2017).

The predictions are then multiplied by a coupling coefficient c, which represents the agreement between capsules. The coupling coefficients c_{ij} between each capsule i in the PrimaryCaps and all the capsules j in the DigitCaps sum to 1 and are determined by a routing softmax function whose initial logits b_{ij} are the log prior probabilities that capsule i should be coupled to capsule j. Hence, c_{ij} is updated iteratively to enable the "Dynamic Routing" process. Eqs. (2)–(5) demonstrate the previous operations.

$$a_{ij} = \mathbf{s}_j \cdot \hat{\mathbf{u}}_{j|i} \tag{2}$$

$$b_{ij} = b_{ij} + a_{ij} \tag{3}$$

$$c_{ij} = \frac{\exp(b_{ij})}{\sum_k \exp(b_{ij})} \tag{4}$$

$$\mathbf{s}_j = \sum_i c_{ij} \hat{\mathbf{u}}_{j|i} \tag{5}$$

where $\mathbf{s}_j$ is weighted sum that is calculated to obtain the candidates for a squashing function v_j. The squashing operation is responsible for generating a normalized vector based on the multiple neurons contained in the capsule. The activation function used in this step is given by Eq. (6).

$$v_j = \frac{\|\mathbf{s}_j\|^2}{1 + \|\mathbf{s}_j\|^2} \cdot \frac{\mathbf{s}_j}{\|\mathbf{s}_j\|} \tag{6}$$

A margin loss function is established to assist the classification process. The function calculates the loss term from the output vector of DigitCaps. This will help in deciding whether the selected digit capsule matches the real target value of class k. The formula of the margin loss function is given by Eq. (7).

$$L_k = T_k \max\left(0, m^+ - \|v_k\|\right)^2 + \lambda\left(1 - T_k\right)\max\left(0, \|v_k\| - m^-\right)^2 \tag{7}$$

Where T_k is a label (0 or 1) indicating whether a class k is present "1" or not "0". Terms m^+, m^-, and λ are the hyperparameters of the model such that $m^+ = 0.9$, $m^- = 0.1$ and $\lambda = 0.5$. The routing procedure can be further described in Algorithm 1.

The original CapsNet performed very well in classifying MNIST handwritten, in which they have archived more than 99% classification accuracy. However, the same network was tested on the well-known CIFAR-10 data and the accuracy dropped down to 90%. The reason behind is these dataset are in RGB format and have more complex background, unlike the simple MNIST data which are in grayscale format. Therefore, the amount of convolutional layers and hyperparameters used in the original CapsNet could not extract all the features in such complex images. In this study, the performance of an improved CapsNet (WaferCaps) that was used to classify WBM defect patterns in Abu Ebayyeh et al. (2022) will be tested and combine its performance with the CNN in Fig. 10. WaferCaps architecture is demonstrated

---

Algorithm 1 Routing algorithm.

1: **procedure** ROUTING($\hat{\mathbf{u}}_{j|i}, r, l$)
2:     for all capsules i in layer l and capsule j in layer $(l+1)$: $b_{ij} \leftarrow 0$.
3:     **for** r iterations **do**
4:         for all capsule i in layer l: $\mathbf{c}_i \leftarrow \texttt{softmax}(\mathbf{b}_i)$
5:         for all capsule j in layer $(l+1)$: $\mathbf{s}_j \leftarrow \sum_i c_{ij}\hat{\mathbf{u}}_{j|i}$
6:         for all capsule j in layer $(l+1)$: $\mathbf{v}_j \leftarrow \texttt{squash}(\mathbf{s}_j)$
7:         for all capsule i in layer l and capsule j in layer $(l+1)$:
8:             $b_{ij} \leftarrow b_{ij} + \hat{\mathbf{u}}_{j|i} \cdot \mathbf{v}_j$
9:         **return** $\mathbf{v}_j$
10:    **end for**
11: **end procedure**

Table 3
Layers structure and parameters used of the proposed WaferCaps for classification and decision fusion approach.

Layer	Type	Input size	Kernel size/stride	Activation	Dropout	Output size
1	conv1	[200, 200, 3]	15/1	ReLU	Yes	[50, 50, 256]
2	conv2	[50, 50, 256]	15/1	ReLU	Yes	[36, 36, 512]
3	conv3	[36, 36, 512]	15/1	ReLU	Yes	[22, 22, 1024]
4	PrimCaps	[22, 22, 1024]	9/2	ReLU	No	[4, 4, 8, 128]
5	WBMCaps	[4, 4, 8, 128]	–	Squash	No	[16, 3]
6	FC	[16, 3]	–	Softmax	No	[3,]

using Fig. 12 and Table 3. As demonstrated in Fig. 12, WaferCaps has two more convolutional layers and increased kernel size, which allows more extensive feature extraction process. Furthermore, dropout layers were used after each convolutional layer to avoid overfitting.

3.3.3. WaferCaps+CNN

A lot of research indicates that a decision fusion approach improves classification performance significantly (Al-Rakhami et al., 2021; Arco et al., 2021; Khaleghi et al., 2013; Li et al., 2021; Nguyen et al., 2020; Singh & Majumder, 2020). Different classification techniques may have varying classification performances for the same problem. Fusion of decisions is a way of integrating the results of multiple classifiers into a common conclusion about an event and has the potential to integrate various decision rules in a fully tunable way. As a result of the decision fusion, classification accuracy can be increased.

There are several types of decision fusion techniques, based on the fusion architecture used (Chandola et al., 2021; Khaleghi et al., 2013):

- Serial decision fusion: One way to implement serial decision fusion is by arranging the classifiers one after another in a series; each classifier's output is used as an input to the next.
- Parallel decision fusion: Two or more classifiers work together in parallel to perform classification simultaneously, and then the classifiers combine the results.
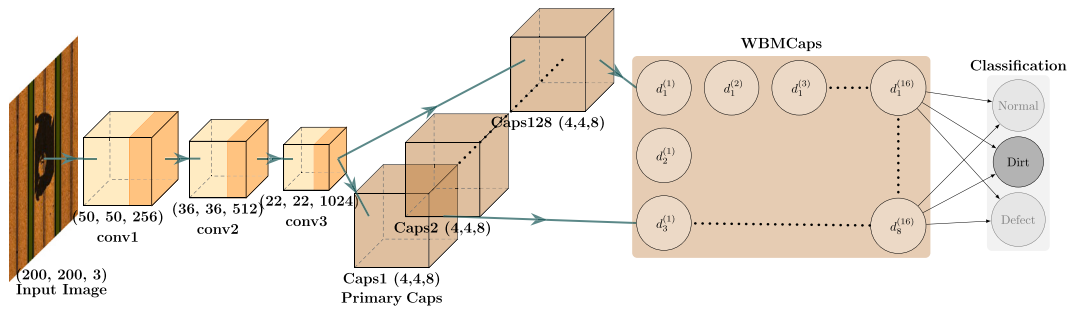
Fig. 12. The proposed WaferCaps for classification and decision fusion approach (Abu Ebayyeh et al., 2022).
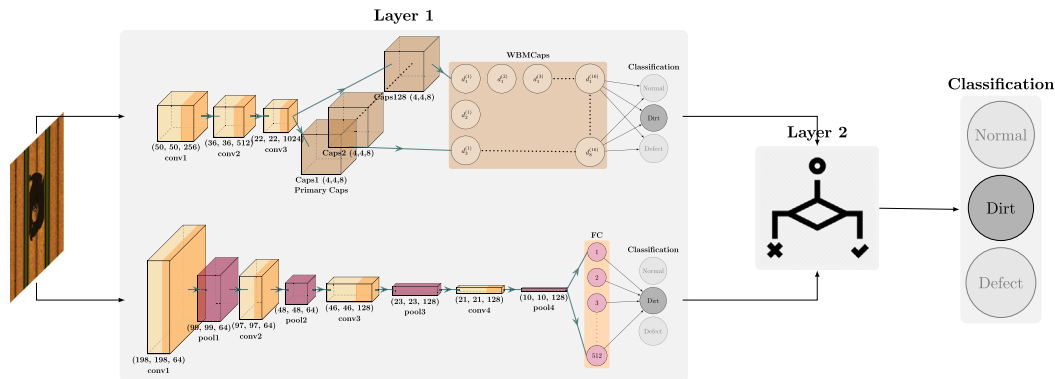

Fig. 13. Architecture of the combined WaferCaps+CNN proposed approach for decision fusion.

• Hybrid decision fusion: This refers to a hierarchical classification process.

In this study, a parallel decision fusion based approach (see Fig. 13) is employed by combining the performance of CNN in Fig. 10 and WaferCaps. As will observed in Section 4 the reason of combining these two classifiers is that WaferCaps performed the best in classifying normal and defect samples, while CNN detected dirt samples more accurately. As a result, the decision fusion approach will increase the final classification accuracy for all the classes. The combined classifier consists of two main layers that are individual classifiers and selection rules.

In the first layer, two individual classifiers are settled, which are built using the same training dataset. The second layer represents the selection methods that receive all individual classifiers' outputs and produces the final result. For every input image, each classifier generates the output that represents the probability of each class. The decimal numbers between 0 and 1 can be interpreted as a percentage of confidence.

In our classifiers, p_1, p_2 and p_3 are probabilities of Defect, Dirt and Normal classes, respectively and $(p_1^{Caps}, p_2^{Caps}, p_3^{Caps})$ and $(p_1^{CNN}, p_2^{CNN}, p_3^{CNN})$ are the outputs of classifiers in the first layers. On one side, the performances of individual classifiers show that WaferCaps classifier has higher sensitivity for Defect and Normal classes than CNN classifier. Therefore, in the next layer, more decision weight is given for WaferCaps when a decision is made for these two classes. On the other side, CNN provides high sensitivity for the Dirt class, thus CNN's output is more reliable.

In the second layer, the selection rules are described in Algorithm 2. Applying these rules to the output of the two classifiers merge the advantages of both classifiers to achieve high accuracy.

4. Results & discussion

In this section, the performance of our proposed combined "WaferCaps+CNN" is evaluated against using non-combined networks and

---

Algorithm 2 Selection rules.

1: **procedure** SELECTION($p_1^{Caps}, p_2^{Caps}, p_3^{Caps}, p_1^{CNN}, p_2^{CNN}, p_3^{CNN}$)
2:    $\left(p_1^{Caps}, p_2^{Caps}, p_3^{Caps}\right)$ ← probabilities of three classes in Wafer-Caps
3:    $\left(p_1^{CNN}, p_2^{CNN}, p_3^{CNN}\right)$ ← probabilities of three classes in CNN
4:    **if** $p_1^{Caps} > 0.8$ **OR** $p_1^{CNN} > 0.95$ **then**
5:      predicted class ← Defect
6:    **else if** $p_2^{CNN} > 0.95$ **then**
7:      predicted class ← Dirt
8:    **else if** $p_3^{CNN} > 0.95$ **then**
9:      predicted class ← Normal
10:    **else**
11:      predicted class ← max(max($p_1^{Caps}, p_2^{Caps}, p_3^{Caps}$), max($p_1^{CNN}, p_2^{CNN}, p_3^{CNN}$))
12:    **end if**
13:    **return** predicted class
14: **end procedure**

---

other DL models. In our evaluation, different metrics were used such as confusion matrices, accuracy, recall, precision, and F1-score. Where the confusion matrix shows the measure of the predicted classes against actual classes. The accuracy predicts how many images were classified correctly. The precision measures the ratio of correctly classified positive samples to the total classified positive samples. The recall is a measure of the ratio of correctly classified positive samples to the all samples in actual class. F1 Score is the weighted average of Precision and Recall. For calculating the previous metrics, according to the confusion matrix, the values of true positive (TP), true negative (TN), false negative (FN) and false positive (FP) should be used as shown in Eqs. (8)–(11).
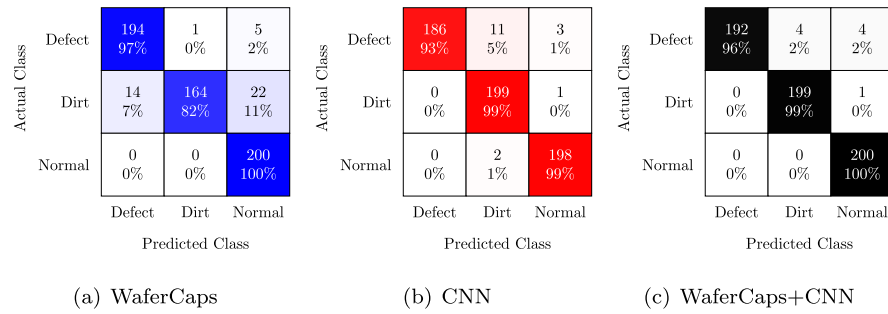
$$Accuracy = \frac{TN + TP}{TN + TP + FN + FP} \tag{8}$$

Fig. 14. Confusion matrix for the validation dataset for different network setups.

$$Precision = \frac{TP}{TP + FP} \qquad (9)$$

$$Recall = \frac{TP}{TP + FN} \qquad (10)$$

$$F1 - score = \frac{2(Precision \times Recall)}{Precision + Recall} \qquad (11)$$

4.1. Experimental data

All data were processed in RGB format with a dimension of 200×200 according to three classes: normal, dirt and defect. The data were divided into two categories; namely, training and validation. In order to investigate the performance of the tested networks, as well as the quality of the augmented data, all of the real samples were used for the dirt and defect classes in the validation dataset, while all the augmented data were used in the training of the networks. The total data samples used were 1,000 images per class. Such that the training dataset consisted of 800 samples, while the validation dataset consisted of 200 samples.

4.2. Proposed method vs. standalone models

A series of experiments were conducted to investigate the performance of using CNN and WaferCaps models separately and compare it with the performance of using the proposed combined model "WaferCaps+CNN". The results of classification for the CNN can be found in the confusion matrix of Fig. 14(b) and metrics in Table 4. It can be observed that CNN achieved higher overall accuracy when compared to WaferCaps, and the performance of the dirt classification was also remarkable in CNN; however, despite of the better overall performance in CNN, WaferCaps achieved better classification results in both defect and normal classes. When both models' performance is combined the overall accuracy is further increased to 98.5%, and it also took the best classification results for the normal and dirt class. However, the classification accuracy for the defect class was slightly dropped when compared to the WafeCaps model

4.3. Proposed method vs. other DL models

The proposed approach was further compared with relevant popular algorithms such as CapsNet, ResNet50, DenseNet, Inception-V3, MobileNet, Xception, decision tree, SVM, k-NN and MLP. The performance of these networks is demonstrated in Fig. 15(j) and Table 4. Comparing the six networks, ResNet50 achieved the best overall accuracy with 93.75%, followed by DenseNet with 92.8%, followed by Inception-V3 with 90%, then Xception with 89.7%, then MobileNet with 89.2%, then CapsNet with 77.2%, then k-NN with 61%, then SVM with 60%, then decision tree with 55.8% and finally MLP with 51%. The proposed approach of "WaferCaps+CNN" outperformed these networks. Fig. 16 compares the proposed approach with all other models used in terms of the misclassified samples for each of them.

Table 4
Metrics for evaluating the validation data against different DL networks.

Model	Overall Accuracy	Metric	Defect	Dirt	Normal
WaferCaps + CNN (proposed)	**0.985**	**Recall**	**0.96**	**0.995**	**1.0**
		Precision	**1.0**	**0.9803**	**0.9756**
		F1-score	**0.9796**	**0.9876**	**0.9876**
WaferCaps	0.93	Recall	0.97	0.82	1.0
		Precision	0.9327	0.9939	0.8811
		F1-score	0.951	0.8986	0.9368
CNN	0.972	Recall	0.93	0.995	0.99
		Precision	1.0	0.9387	0.9802
		F1-score	0.9637	0.966	0.9851
CapsNet	0.772	Recall	0.725	0.725	0.865
		Precision	0.7214	0.8841	0.7362
		F1-score	0.7232	0.7967	0.7954
ResNet50	0.9375	Recall	0.925	0.9	0.995
		Precision	0.9788	0.9375	0.9087
		F1-score	0.9512	0.9184	0.9499
DenseNet	0.928	Recall	0.875	0.91	1.0
		Precision	1.0	0.9234	0.8772
		F1-score	0.9333	0.9169	0.9346
Inception-V3	0.9	Recall	0.925	0.775	1.0
		Precision	0.984	0.9226	0.8197
		F1-score	0.9536	0.8424	0.901
Xception	0.897	Recall	0.845	0.895	0.95
		Precision	0.8711	0.8364	0.9896
		F1-score	0.8579	0.8647	0.9693
MobileNet	0.892	Recall	0.975	0.88	0.82
		Precision	0.8125	0.9944	0.8962
		F1-score	0.8864	0.9337	0.8564
k-NN	0.61	Recall	1.0	0.0	0.83
		Precision	0.463	0.0	1.0
		F1-score	0.633	0.0	0.9071
SVM	0.6	Recall	0.81	0.0	1.0
		Precision	1.0	0.0	0.46
		F1-score	0.9	0.0	0.63
Decision Tree	0.558	Recall	1.0	0.0	0.675
		Precision	0.463	0.0	1.0
		F1-score	0.633	0.0	0.806
MLP	0.51	Recall	0.68	0.66	0.19
		Precision	0.4772	0.5176	0.6333
		F1-score	0.5608	0.5802	0.2923

5. Conclusion

Development of industry ready AOI solution to classify waveguide anomalies in QCL wafers according to three classes (defect, dirt and normal) was the aim of this research. In order to achieve this, a mechanism of image acquisition, data augmentation and DL classifier were proposed that combines two algorithms (WaferCaps and
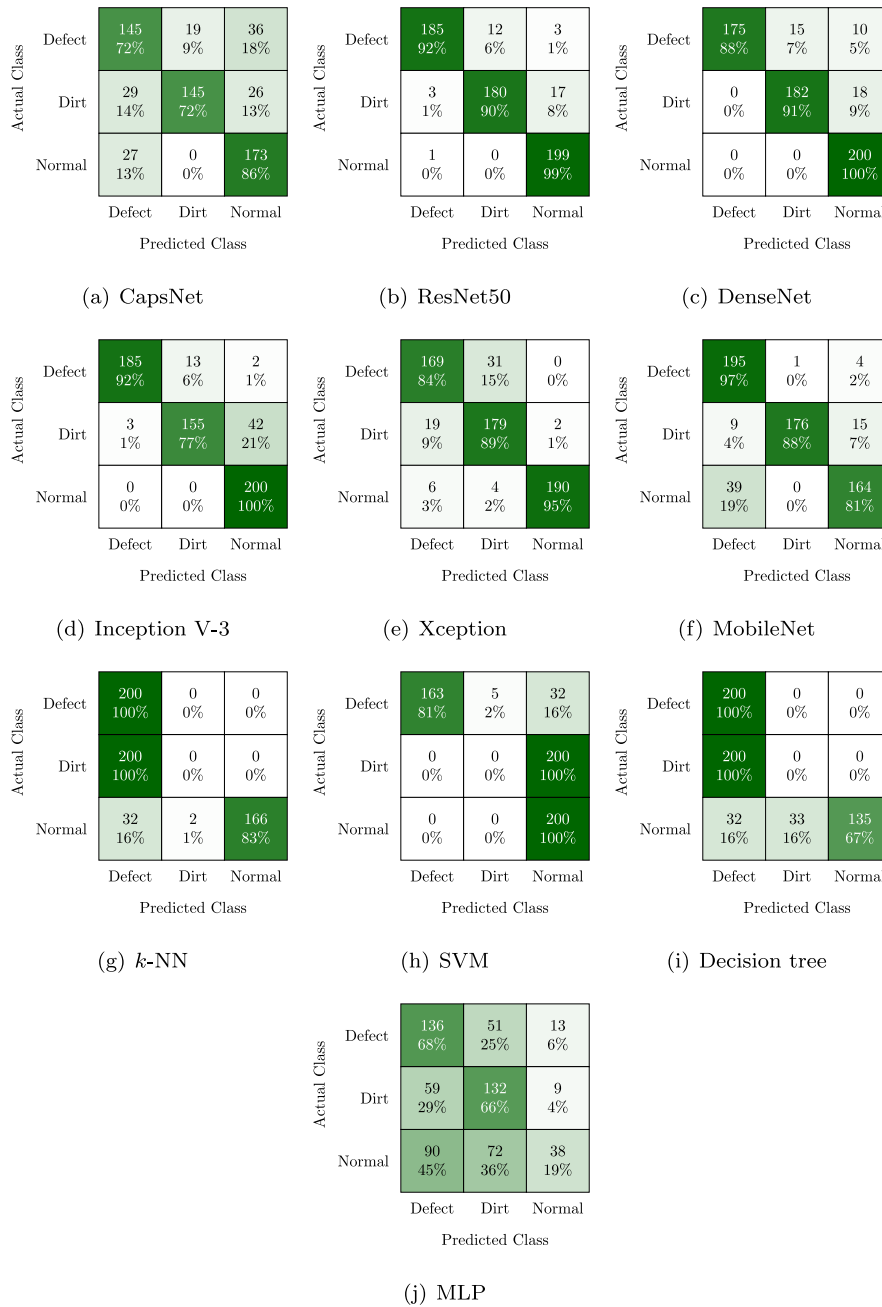
Fig. 15. Confusion matrix for the validation dataset for different network setups.

CNN) using parallel decision fusion. The customized image acquisition system consists of a microscope and camera. The wafer images were then segmented into multiple samples, such that each sample has 200 × 200 of size in RGB format. Data augmentation approach was then used to increase the samples of the dirt and defect anomalies in a way that 1,000 images were used in each category. The total dataset was manually labeled by experts. All the generated dataset was used for training, and the real samples were used for validation.

The proposed DL classifier (WaferCaps+CNN) combines the performance of WaferCaps and CNN by rule-based selection of the predictions generated. It showed accurate and robust capability to detect QCL anomalies with high validation accuracy of 98.5%. The performance of the proposed solution was compared with performance of using standalone CNN and WaferCaps as well as with state-of-the-art DL

such as CapsNet, ResNet-50, Inception-V3, DenseNet, MobileNet and Xception and other wel-known ML algorithms such as SVM, decision tree, k-NN and MLP. It was proven that the overall performance of WaferCaps+CNN outperformed them all.

Despite the remarkable achievement of identifying normal, dirt and defect samples accurately, this research has some limitations. First, in case of the change of production methods, other possible anomalies and defects feature may rise, which cannot be identified by our approach that is trained to classify three classes. This may open the door on using uncertainty learning. Second, augmenting samples manually is time consuming and may be not the best way in dealing with big data. Therefore, more samples will be acquired in the future to use more effective methods in data augmentation such as DCGAN. Finally, the algorithm cannot specify the location of the anomaly and can only classify the sample. In a future research, more samples will be used and
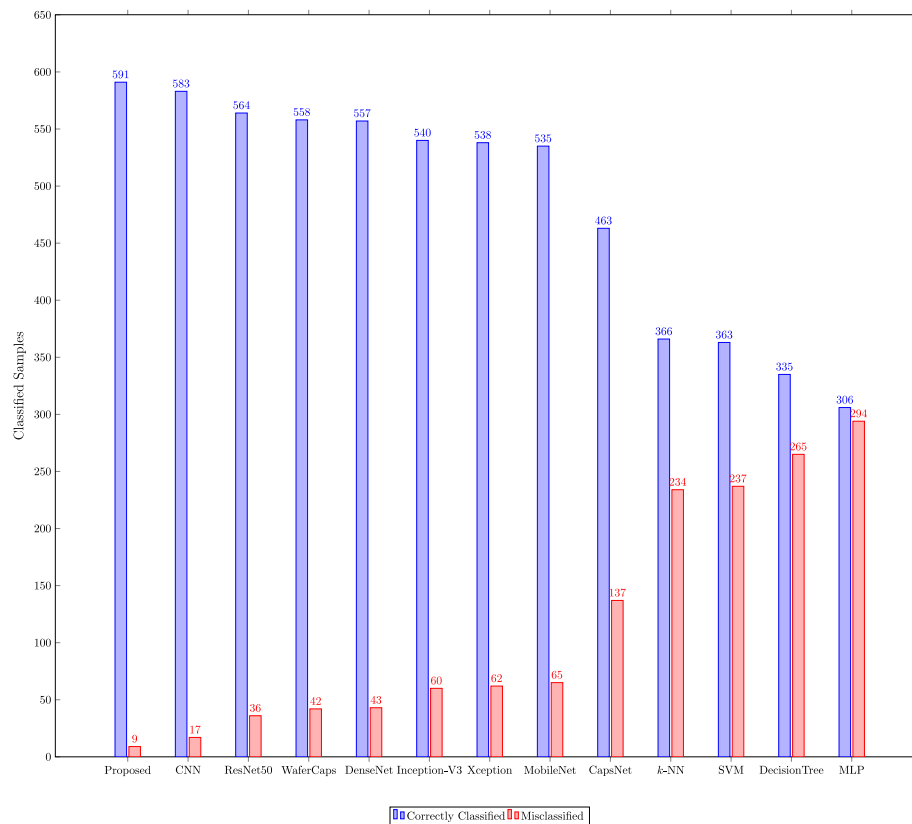
Fig. 16. Classified vs. misclassified samples for proposed approach against other models.

a dedicated algorithm to specify the location of the anomaly accurately will be proposed.

CRediT authorship contribution statement

Abd Al Rahman M. Abu Ebayyeh: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing. **Alireza Mousavi:** Conceptualization, Supervision. **Sebelan Danishvar:** Conceptualization, Methodology, Software, Validation, Formal analysis. **Stéphane Blaser:** Data acquisition, Handling, Software. **Tobias Gresch:** Data acquisition, Handling, Software. **Olivier Landry:** Data acquisition, Handling, Software. **Antoine Müller:** Data acquisition, Handling, Software.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that has been used is confidential.

Funding

References

Abdiyeva, K., Tibeyev, T., & Lukac, M. (2020). Capacity limits of fully binary CNN. In *2020 IEEE 50th International symposium on multiple-valued logic* (pp. 206–211). http://dx.doi.org/10.1109/ISMVL49045.2020.000-4.

Abu Ebayyeh, A. M., Danishvar, S., & Mousavi, A. (2022). An improved capsule network (WaferCaps) for wafer bin map classification based on DCGAN data upsampling. *IEEE Transactions on Semiconductor Manufacturing*, *35*(1), 50–59. http://dx.doi.org/10.1109/TSM.2021.3134625.

Abu Ebayyeh, A. M., & Mousavi, A. (2020). A review and analysis of automatic optical inspection and quality monitoring methods in electronics industry. *IEEE Access*, *8*, 183192–183271. http://dx.doi.org/10.1109/access.2020.3029127.

Al-Rakhami, M. S., Islam, M. M., Islam, M. Z., Asraf, A., Sodhro, A. H., & Ding, W. (2021). Diagnosis of COVID-19 from X-rays using combined CNN-RNN architecture with transfer learning. http://dx.doi.org/10.1101/2020.08.24.20181339, MedRxiv.

Arco, J. E., Ramírez, J., Górriz, J. M., & Ruz, M. (2021). Data fusion based on searchlight analysis for the prediction of alzheimer's disease. *Expert Systems with Applications*, *185*, Article 115549. http://dx.doi.org/10.1016/j.eswa.2021.115549.

Cha, Y. J., Choi, W., & Büyüköztürk, O. (2017). Deep learning-based crack damage detection using convolutional neural networks. *Computer-Aided Civil and Infrastructure Engineering*, *32*(5), 361–378. http://dx.doi.org/10.1111/mice.12263.

Chandola, Y., Virmani, J., Bhadauria, H., & Kumar, P. (2021). Chapter 4 - end-to-end pre-trained CNN-based computer-aided classification system design for chest radiographs. In *Primers in Biomedical Imaging Devices and Systems*, *Deep learning for chest radiographs* (pp. 117–140). Academic Press: http://dx.doi.org/10.1016/B978-0-323-90184-0.00011-4.

Chien, J.-C., Wu, M.-T., & Lee, J.-D. (2020). Inspection and classification of semiconductor wafer surface defects using CNN deep learning networks. *Applied Sciences*, *10*(15), 5340. http://dx.doi.org/10.3390/app10155340.

Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions. In *2017 IEEE Conference on computer vision and pattern recognition* (pp. 1800–1807). http://dx.doi.org/10.1109/CVPR.2017.195.

Cruz, Y. J., Rivas, M., Quiza, R., Villalonga, A., Haber, R. E., & Beruvides, G. (2021). Ensemble of convolutional neural networks based on an evolutionary algorithm applied to an industrial welding process. *Computers in Industry*, *133*, Article 103530. http://dx.doi.org/10.1016/j.compind.2021.103530.

Curl, R. F., Capasso, F., Gmachl, C., Kosterev, A. A., McManus, B., Lewicki, R., Pusharsky, M., Wysocki, G., & Tittel, F. K. (2010). Quantum cascade lasers in chemical physics. *Chemical Physics Letters*, *487*(1–3), 1–18. http://dx.doi.org/10.1016/j.cplett.2009.12.073.

Danishvar, M., Danishvar, S., Souza, F., Sousa, P., & Mousavi, A. (2021). Coarse return prediction in a cement industry's closed grinding circuit system through a fully connected deep neural network (FCDNN) model. *Applied Sciences*, *11*(4), http://dx.doi.org/10.3390/app11041361.

Edinbarough, I., Balderas, R., & Bose, S. (2005). A vision and robot based on-line inspection monitoring system for electronic manufacturing. *Computers in Industry*, *56*(8–9), 986–996. http://dx.doi.org/10.1016/j.compind.2005.05.022.

Faist, J., Capasso, F., Sivco, D. L., Sirtori, C., Hutchinson, A. L., & Cho, A. Y. (1994). Quantum cascade laser. *Science*, *264*(5158), 553–556. http://dx.doi.org/10.1126/science.264.5158.553.

Figueiredo, P., Suttinger, M., Go, R., Tsvid, E., Patel, C. K. N., & Lyakh, A. (2017). Progress in high-power continuous-wave quantum cascade lasers. *Applied Optics*, *56*(31), H15–H23. http://dx.doi.org/10.1364/ao.56.000h15.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *2016 IEEE Conference on computer vision and pattern recognition* (pp. 770–778). http://dx.doi.org/10.1109/CVPR.2016.90.

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., & Adam, H. (2017). MobileNets: Efficient convolutional neural networks for mobile vision applications. CoRR abs/1704.04861arXiv:1704.04861.

Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In *2017 IEEE Conference on computer vision and pattern recognition* (pp. 2261–2269). http://dx.doi.org/10.1109/CVPR.2017.243.

Huang, S.-H., & Pan, Y.-C. (2015). Automated visual inspection in the semiconductor industry: A survey. *Computers in Industry*, *66*, 1–10. http://dx.doi.org/10.1016/j.compind.2014.10.006.

Iglesias, C., Martínez, J., & Taboada, J. (2018). Automated vision system for quality inspection of slate slabs. *Computers in Industry*, *99*, 119–129. http://dx.doi.org/10.1016/j.compind.2018.03.030.

Khaleghi, B., Khamis, A., Karray, F. O., & Razavi, S. N. (2013). Multisensor data fusion: A review of the state-of-the-art. *Information Fusion*, *14*(1), 28–44. http://dx.doi.org/10.1016/j.inffus.2011.08.001.

Kim, M. H., Weyrich, T., & Kautz, J. (2009). Modeling human color perception under extended luminance levels. *ACM Transactions on Graphics*, *28*(3), 1–9. http://dx.doi.org/10.1145/1531326.1531333.

Li, Y., Wang, H., Yang, J., Wang, K., & Qi, G. (2021). A non-intrusive load monitoring algorithm based on multiple features and decision fusion. *Energy Reports*, *7*, 1555–1562. http://dx.doi.org/10.1016/j.egyr.2021.09.087, 2021 International Conference on Energy Engineering and Power Systems.

Liu, E., Chen, K., Xiang, Z., & Zhang, J. (2019). Conductive particle detection via deep learning for ACF bonding in TFT-LCD manufacturing. *Journal of Intelligent Manufacturing*, *31*(4), 1037–1049. http://dx.doi.org/10.1007/s10845-019-01494-9.

Mei, S., Yang, H., & Yin, Z. (2018). An unsupervised-learning-based approach for automated defect inspection on textured surfaces. *IEEE Transactions on Instrumentation and Measurement*, *67*(6), 1266–1277. http://dx.doi.org/10.1109/tim.2018.2795178.

Muzammal, M., Talat, R., Sodhro, A. H., & Pirbhulal, S. (2020). A multi-sensor data fusion enabled ensemble approach for medical data from body sensor networks. *Information Fusion*, *53*, 155–164.

Nguyen, T. T., Dang, M. T., Baghel, V. A., Luong, A. V., McCall, J., & Liew, A. W.-C. (2020). Evolving interval-based representation for multiple classifier fusion. *Knowledge-Based Systems*, *201–202*, Article 106034. http://dx.doi.org/10.1016/j.knosys.2020.106034.

Pecharroman-Gallego, R. (2017). An overview on quantum cascade lasers: Origins and development. *Quantum Cascade Lasers*, http://dx.doi.org/10.5772/65003.

Portalés, C., & Ribes-Gómez, E. (2015). An image-based system to preliminary assess the quality of grape harvest batches on arrival at the winery. *Computers in Industry*, *68*, 105–115. http://dx.doi.org/10.1016/j.compind.2014.12.010.

Razeghi, M. (2009). High-performance InP-based mid-IR quantum cascade lasers. *IEEE Journal of Selected Topics in Quantum Electronics.*, *15*(3), 941–951. http://dx.doi.org/10.1109/jstqe.2008.2006764.

Sabour, S., Frosst, N., & Hinton, G. E. (2017). Dynamic routing between capsules. In *NIPS'17, Proceedings of the 31st International conference on neural information processing systems* (pp. 3859–3869). Red Hook, NY, USA: Curran Associates Inc..

Singh, S. A., & Majumder, S. (2020). Chapter one - short and noisy electrocardiogram classification based on deep learning. In *Deep learning for data analytics* (pp. 1–19). Academic Press: http://dx.doi.org/10.1016/B978-0-12-819764-6.00002-8.

Sodhro, A. H., S.Obaidat, M., Pirbhulal, S., Sodhro, G. H., Zahid, N., & Rawat, A. (2019). A novel energy optimization approach for artificial intelligence-enabled massive internet of things. In *2019 International symposium on performance evaluation of computer and telecommunication systems* (pp. 1–6). http://dx.doi.org/10.23919/SPECTS.2019.8823317.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on computer vision and pattern recognition* (pp. 2818–2826). http://dx.doi.org/10.1109/CVPR.2016.308.

Villa, N., Strübi, G., Gresch, T., Butet, J., Blaser, S., & Müller, A. (2019). Quantum cascade lasers with discrete and non equidistant extended tuning tailored by simulated annealing. *Optics Express*, *27*(19), 26701. http://dx.doi.org/10.1364/oe.27.026701.

Wang, M.-J., & Huang, C.-L. (2004). Evaluating the eye fatigue problem in wafer inspection. *IEEE Transactions on Semiconductor Manufacturing*, *17*(3), 444–447. http://dx.doi.org/10.1109/tsm.2004.831943.

Westphal, E., & Seitz, H. (2021). A machine learning method for defect detection and visualization in selective laser sintering based on convolutional neural networks. *Additive Manufacturing*, *41*, Article 101965. http://dx.doi.org/10.1016/j.addma.2021.101965.

Yang, H., Chen, Y., Song, K., & Yin, Z. (2019). Multiscale feature-clustering-based fully convolutional autoencoder for fast accurate visual inspection of texture surface defects. *IEEE Transactions on Automation Science and Engineering*, *16*(3), 1450–1467. http://dx.doi.org/10.1109/tase.2018.2886031.

Yu, J., Zheng, X., & Liu, J. (2019). Stacked convolutional sparse denoising auto-encoder for identification of defect patterns in semiconductor wafer map. *Computers in Industry*, *109*, 121–133. http://dx.doi.org/10.1016/j.compind.2019.04.015.