PROCESSES

Performance evaluation of CNN and R-CNN based line by line analysis algorithms for fibre placement defect classification

Sebastian Meister[1] · **Mahdieu Wermes**[1]

Abstract

The *Automated Fibre Placement* process is commonly used in aerospace for the manufacturing of structural components, but requires a subsequent inspection to meet the corresponding safety requirements. In order to improve this mostly manual inspection step, machine learning methods for the interpretation of 2D surface images are being increasingly utilised in research. Depending on the manufacturing process, a very long time can elapse between the appearance of a manufacturing defect and its recognition. Hence, in this paper *Convolutional* and *Recurrent Neural Network* techniques are presented that allow a line-by-line analysis of the incoming height profile scans of a *Laser Line Scan Sensor* as a 1D signal, which enables a direct reaction to a defect, even if only one or a few individual height profiles of the defect have been recorded. The combination of *Convolutional* and *Recurrent Neural Network* structures is particularly beneficial for this application. The investigations in this paper are especially interesting for developers of automated inspection systems in composite engineering.

1 Introduction

Lightweight structures are widely used in the aerospace industry, for instance in the Airbus A350 XWB or the Boeing 787 [1, 2]. The Carbon Fibre Reinforced Plas- tic (CFRP) material often have a greater stiffness and strength than metallic materials, which makes them ideal for lightweight structures. The production of these usually complex structures is often quite expensive, which is why fast and efficient manufacturing techniques are essential to realise an economical production. To ensure the quality of the components, an additional visual inspection is carried out in the aerospace industry after manufacturing.

This visual inspection often takes up to 50% [3] of the overall manufacturing time and it is also often very difficult to achieve the required inspection quality, which offers great potential for improvement.

An important step in automated inspection is the reliable classification of manufacturing defects from sensor data [4, 5]. Machine learning methods are particularly well-suited for this purpose, although currently mostly the entire 2D images of a material surface are evaluated [6, 7]. This approach involves two challenges: Firstly, a large number of full training images are required to train modern Neural Network based classifiers [8–10], and secondly, a full image of the defect must be recorded before a classification and subsequent system response can be performed [11–13]. Thus, depending on the size of the captured image and the speed of the manufacturing process, considerable time can elapse between the appearance of a manufacturing defect and its classification. To address this issue, it has been investigated approaches to classify fibre layup defects from typical Automated Fibre Placement (AFP) processes through line-by-line interpretation of the input data. On the one hand, this offers the potential to reduce the time between the defect appearance and classification, and on the other hand, it offers the possibility to provide significantly more training data.

For the investigations in this study, the application case of AFP manufacturing is considered, as this is a widely used process in industry and thus the transferability of the research results is ensured [14–17]. A Laser Line Scan

✉ Sebastian Meister
   sebastian.meister@dlr.de

1 Center for Lightweight Production Technology (ZLP), German Aerospace Center (DLR), Ottenbecker Damm 12, Stade, Germany

Sensor (LLSS) is often applied for inline inspection in this process, which is why data from such a sensor system is used in this research [11, 15, 18, 19]. This sensor type projects a laser line onto a surface and calculates the surface topology line by line from the reflected beam.

[20] The research question addressed in this publication is:

Which Artificial Neural Network (ANN) architectures are suitable to perform a line-by-line interpretation of LLSS height profile scans with respect to AFP fibre placement defects?

The structure of this paper involves a description of the state of the art and research, followed by the development of suitable ANN architectures. Then the performance of the various ANN designs are examined for real and synthetically generated data of fibre layup defects.

2 Related research

2.1 Manufacturing process

Various fibre placement methods are available whereby *Automated Fibre Placement* (AFP) [21, 22], Dry Fibre Placement (DFP) [21, 22], Automated Tape Laying (ATL) [21] and Direct Roving Placement (DRP) [23] are particularly popular techniques. They apply CFRP material in layers to a mould, which is illustrated in Fig. 1 and described in detail by Campbell [24].

For the fabrication of more complex lightweight components, the AFP process is often used, which lays down narrow strips of fibre material along a given path [17, 24, 25]. This involves an effector guiding the fibre material to the mould's surface where it is heated up and pressed onto the mould or layer underneath [21, 24]. Rudberg [26] expects an increasing application of this flexible manufacturing process in the future.

During AFP, various defects can arise which reduce the mechanical properties of a component and are therefore undesirable [17, 27].Typical defect types from the literature are wrinkles, twists, foreign bodies, overlaps and gaps, which are visualised in Fig. 2.

2.2 Composite inspection and data

The monitoring of AFP processes is being increasingly investigated in industry and research. InFactory Solutions [15], Electroimpact [16, 28], Danobat Composites [29] and Profactor [30] use LLSS for AFP monitoring, which captures the 3D topology of the deposited composite material.

Sacco et al. [18, 31] studied the Convolutional Neural Network (CNN) based segmentation of defects from LLSS depth images of AFP layup defects. They utilised 800×800 pixel LLSS depth images for training their fully linked CNN. Zambal et al. [10, 32] presented an end-to-end deep learning approach for AFP defect segmentation under consideration of synthesised training images. For this purpose, they used a U-net CNN introduced by Ronneberger et al. [33] in 2015 along with realistic validation images from a LLSS.

In previous research Meister et al. addresses the segmentation of fibre layup defects in full scan images [7, 11], the synthetic generation of training images for the AFP inspection [12] as well as the robust and traceable classification of such defects [4, 5, 34]. For the investigations in this paper, the real defect images and synthetic data from the previous papers [7, 11, 12] are utilised. For recording the real scan images, manually produced defect samples are scanned with a LLSS. The defect samples are made of 1/4" prepreg tows, with one or more defects located on a single defect sample. For the scanning process, the LLSS is attached to an articulated robot, which moves it parallel to the composites surface at a speed of 200 mm/s. The image data recorded in this way is then smoothed using the Contrast Limited Adaptive Histogram Equalization (CLAHE) algorithm. Subsequently, the synthetic defect images are generated on the basis of the real defects via a Deep Convolutional Generative Adversarial Network (DCGAN) algorithm in order to obtain a significantly larger data base. The studies from Meister et al. [7, 11] describe in detail the acquisition and pre-processing of the real defect images, whereas the paper [12] outlines the synthesis of the synthetic defect images additionally applied in this study. Randomly selected images from the real defect
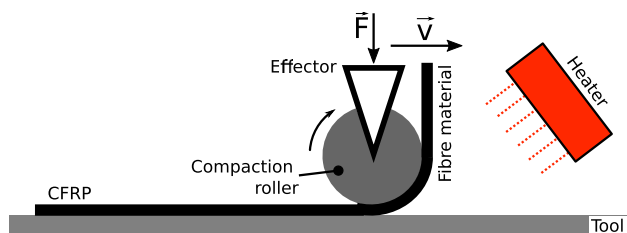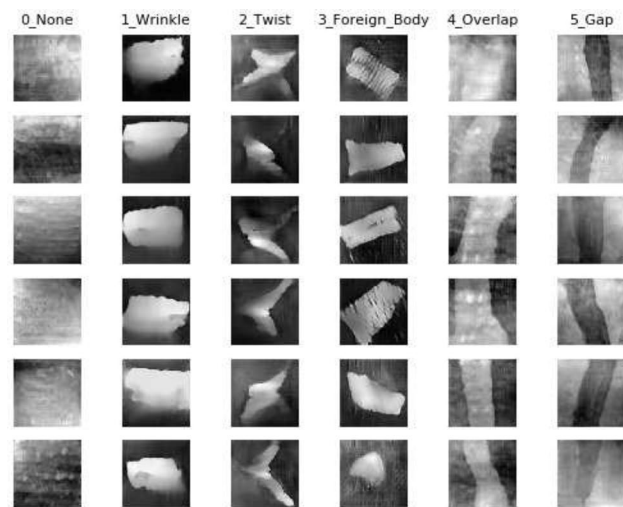

Fig. 1 AFP process with a heater and a compaction roller. Compaction force: **F**; Effector velocity: **v**
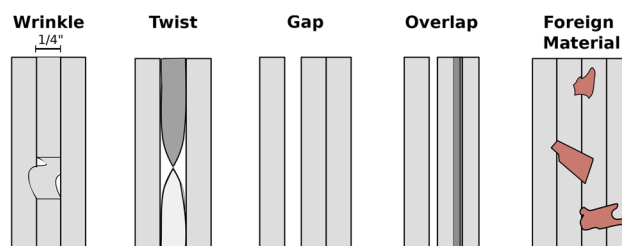
Fig. 2 Common AFP defects
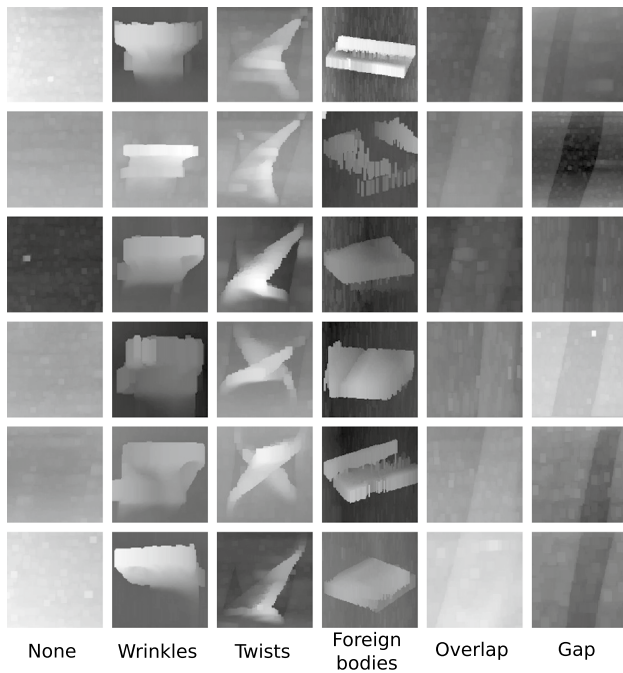
Fig. 3 Randomly chosen real defect depth maps captured by a LLSS for different classes, each having the dimension of 128×128 px from the previous paper Meister et al. [11, 12]

Fig. 4 Randomly chosen synthetic defect depth maps for different classes, each having the dimension of 128×128 px from the previous paper Meister et al. [12]

dataset are illustrated as examples in Fig. 3 and from the synthetic dataset in Fig. 4.

In addition to CNNs, Recurrent Neural Network (RNN) architectures are often applied to evaluate time series data, for instance to predict faults or identify changes in sequential data [35–37].Such techniques are explained in more detail in the following section.

2.3 Recurrent neural networks

RNNs are ANNs which are dedicated to process sequential data. It processes a sequence of data vectors $\mathbf{x}_t = (\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T)$ with time step t, where in reality the RNN works on individual mini-batches [38].In actual operation, gated RNNs such as Long Short-Term Memory (LSTM) or Gated Recurrent Unit (GRU) are frequently used. A fundamental idea of LSTM is the ability to forget an older state of an entry by so-called self-loops. This is implemented by internal recurrence in individual LSTM cells. Essential components of these LSTM cells are the state unit $\mathbf{s}_{t,i}$ and the forget gate $\mathbf{f}_{t,i}$ which adjusts the self-loop weights and can be formulated as [35, 38, 39]:

$$\mathbf{f}_{t,i} = \sigma\left(b_i^f + \sum_j U_{i,j}^f \mathbf{x}_{t,j} + \sum_j W_{i,j}^f \mathbf{h}_{t-1,j}\right) \quad (1)$$

where $\mathbf{x}_t$ is the input data and $\mathbf{h}_t$ the hidden layer vector with all outputs of the individual LSTM cells. Moreover W^f are recurrent weights, U^f the input weights and b^f represent respective biases.

The internal LSTM long-term state $\mathbf{s}_{t,i}$ is updated with the conditional self-loop weight $\mathbf{f}_{t,i}$ as follows [35, 38, 39]:

$$\mathbf{s}_{t,i} = \mathbf{f}_{t,i}\mathbf{s}_{t-1,i} + \mathbf{g}_{t,i}\sigma\left(b_i + \sum_j U_{i,j}\mathbf{x}_{t,j} \right.$$
$$\left. + \sum_j W_{i,j}\mathbf{h}_{t-1,j}\right) \quad (2)$$

with $\mathbf{g}_{t,i}$ as external input gate unit which is calculated analogous to the forget gate in Equation 1. The output $\mathbf{h}_{t,i}$ of a hidden LSTM cell, which represents the short-term state, can also be switched off via the output gate $\mathbf{q}_{t,i}$ using a sigmoid function, which can be expressed as [35, 38, 39]:

$$\mathbf{h}_{t,i} = \tanh(\mathbf{s}_{t,i})\mathbf{q}_{t,i} \quad (3)$$

$\mathbf{q}_{t,i}$ is again calculated analogous to the forget gate from Eq. 1 with it's own parameters W^0, U^0 and b^f.

The other very popular method is GRU, as stated above. The main difference to LSTM is, that GRU has a single gating unit that controls both the forgetting factor as well as the update decision and there is no distinction between short- and long-term state:

$$\mathbf{h}_{t,i} = \mathbf{u}_{t-1,i}\mathbf{h}_{t-1,i} + (1 - \mathbf{u}_{t-1,i})\sigma\left(b_i \sum_j U_{i,j}\mathbf{x}_{t-1,j} \right.$$
$$\left. + \sum_j W_{i,j}\mathbf{r}_{t-1,j}\mathbf{h}_{t-1,j}\right) \quad (4)$$

with $\mathbf{u}_t$ representing the update gate and $\mathbf{r}_{t-1,j}$ the reset gate [38, 40].

3 Methodology

3.1 Experimental data and processing

For the investigations, suitable defect types were selected with reference to Sect. 2.1. Accordingly, flawless areas, wrinkles, twists, foreign bodies, gaps and overlaps are examined. These defect types have already been used in related studies by Oromiehie et al. [17], Harik et al. [27] as well as Heinecke and Willberg [41] and are therefore a suitable choice for the investigations in this paper. As mentioned above, the basis for this paper was the real and synthetic data from previous studies [7, 11, 12] which are illustrated in Figs. 3 and 4. The number and distribution of the available data is summarised in Table 1 once again.

As explained in Meister et al. [12], the synthetic data was generated using a conditional DCGAN. The applied DCGAN consists of a generator with six convolutional layers and a discriminator with five convolutional layers. The

processing was performed on a machine with Intel Xeon Gold 5122 @ 3.60 GHz CPU, 48 GB RAM and a NVIDIA Quadro P6000 GPU.

3.2 Analysis concept of images

The idea behind this research was to analyse individual image rows in order to classify a fibre placement defect. Thus, instead of examining an entire image, as often described in the related research from Sect. 2.2, only a single image row was considered. The signal that was derived from each individual image row represents the grey scale values along the pixels of that respective row. This is illustrated in Fig. 5 using a twist defect as an example. On the left side in Fig. 5 a full defect image is displayed, from which in this paper only the individual signals per row were used for processing, as presented on the right side in Fig. 5b.

The idea behind was, based on the modelling from Meister et al. [20], that a single height profile of the LLSS, which represents the height profile of the surface along the laser line at a certain point in time, can also be understood as a sequence of point by point scans of the defect's height profile. Therefore, the laser line, which was recorded at a given

Table 1 The number of available real and synthetic defect images used for the experiments in this study is given

Data	None	Wrinkle	Twist	Foreign body	Overlap	Gap
Real defs.	86	49	53	22	166	93
Synth. defs.	1000	1000	1000	1000	1000	1000
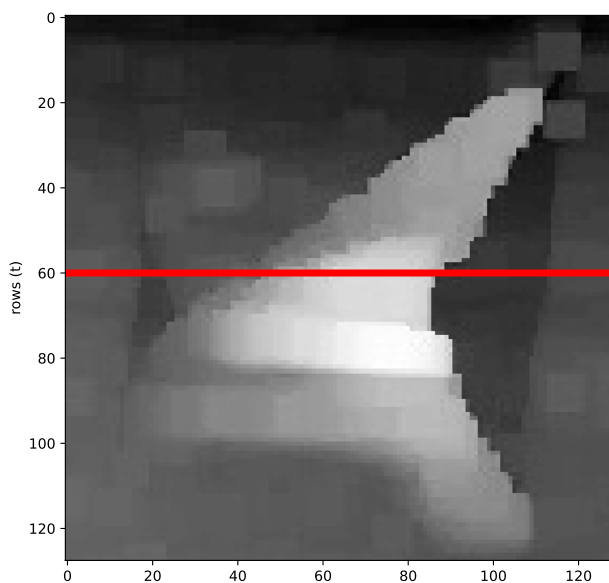
(a) Real twist defect evaluated along the red line.
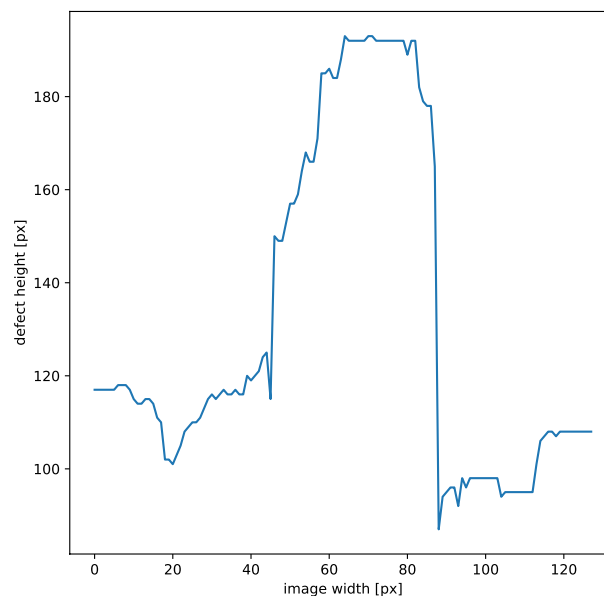
(b) Height profile along the red line.

Fig. 5 A twist defect and the respective height profile along the read line are presented

Table 2 Overview of hyper-parameters.

Parameter	Value
Image size	$128 \times 128 \times 1$
Batch size	32
Optimiser	Adam
Learning rate	0.0002
β_1	0.5
β_2	0.999
ϵ	10^{-7}
Activation functions	ReLU
Training epochs	40

β_i give the exponential decay rate for the i-th moment estimates. ϵ describes a constant for numerical stability

point in time, was interpreted as a sequence of individual measuring points at neighbouring measuring positions. Each pixel along the laser line defines an inherent sampling point with a small positional distance to the previous pixel.

3.3 Neural network setups

In accordance with the analysis idea outlined in Sect. 3.2, ANN architectures for the line-by-line interpretation of defect images are presented below. Referring to the CNN architecture from the previous studies by Meister et al. [4, 5, 12], a 1D CNN architecture was first constructed. The design is directly inspired by these studies and adapted to a 1D case. Subsequently, recurrent designs have been set up, where the input part is again designed as stacked convolutional layers followed by LSTM or GRU units which have been introduced in Sect. 2.3. Since it was very time consuming to determine exactly the number of recurrent units required and adds only limited value to the very special case of fibre placement defect detection, the number of recurrent units was changed with a larger step size. Thus, the performance of the RNN was investigated for 50, 100 and 200 recurrent units respectively.

General parameters that were consistently used across all experimental setups are listed in Table 2.

The first network architecture to be used as the 1D CNN is given in Table 3, where the configurations and dimensions of the individual layers are indicated. The column 'output shape' indicates the size of the output data after each individual layer, where 'None' in this case means that the dimension is variable, depending on the configuration of the Neural Network.

Table 4 presents the layered structure including configuration of the LSTM and GRU.

Table 5 provides an overview of the total number of network parameters for the individual architectures and describes the amount of trainable and non-trainable

Table 3 The layer architecture and number of parameters for the applied 1D CNN are presented

Layer type	Config.	Output shape
Conv1D	$s_k = 5$	(None, 128, 32)
Conv1D	$s_k = 5$	(None, 128, 64)
MaxPooling1D	$s_p = 2$	(None, 64, 64)
Batch normalization	$mom = 0.8$	(None, 64, 64)
Conv1D	$s_k = 5$	(None, 64, 128)
Conv1D	$s_k = 5$	(None, 64, 256)
MaxPooling1D	$s_p = 2$	(None, 32, 256)
Batch normalization	$mom = 0.8$	(None, 32, 256)
Flatten	–	(None, 8192)
Dense	–	(None, 100)
Dropout	0.15	(None, 100)
Batch normalization	$mom = 0.99$	(None, 100)
Dense	–	(None, 6)

This design is related to Meister et al. [4, 5, 12, 13]. s_k: kernel size; s_p: pooling size; mom: momentum

Table 4 Layer architecture for applied RNN approaches {LSTM, GRU} with given number of units #UNITS {50, 100, 200}.

Layer type	Config.	Output shape
Conv1D	$s_k = 5$	(None, 128, 32)
Conv1D	$s_k = 5$	(None, 128, 64)
Conv1D	$s_k = 3$	(None, 128, 128)
Conv1D	$s_k = 3$	(None, 128, 256)
Conv1D	$s_k = 15$	(None, 128, 512)
Dropout	0.5	(None, 128, 512)
MaxPooling1D	$s_p = 2$	(None, 64, 512)
Recurrent Unit	{LSTM, GRU}	(None, <#UNITS>)
Dropout	0.5	(None, <#UNITS>)
Dense	-	(None, 100)
Dense	-	(None, 6)

s_k kernel size, s_p pooling size, mom momentum

Table 5 Overview trainable parameters

	Trainable	Non-trainable	Total
1D CNN	1,036,426	840	1,037,266
LSTM 50	2,218,658	0	2,218,658
LSTM 100	2,356,258	0	2,356,258
LSTM 200	2,691,458	0	2,691,458
GRU 50	2,186,498	0	2,186,498
GRU 100	2,291,098	0	2,291,098
GRU 200	2,545,298	0	2,545,298

parameters. The non-trainable parameters are constant settings which were not updated during the training, whereas the trainable parameters are slightly modified after each iteration in the training process. This gives an impression of the network's complexity and allows conclusions about possible overfitting issues.

3.4 Performance validation

In order to investigate the performance of the individual ANN architectures, the real image data of the height profile scans of the fibre placement defects were initially taken. This was then divided in half into a training and a test data set and then separated into individual lines to be used for the analysis. In this way, a proper separation of the training and test data sets was achieved. For the following investigations, the test data was again divided into validation data, which was used for validation during the training, and test data, which was finally taken in order to analyse the performance of an approach. This results in a final data split of 50% training data, 25% validation data and 25% test data. This split was chosen in particular because in some cases only very few truly different real data sets per class were available, as indicated in Table 1. In order to provide sufficient data for a robust training process, but still allow a meaningful validation/ testing, the distribution given above was chosen.

The investigations were first carried out for the setups outlined above in Sect. 3.3. The accuracy and the loss for the training and validation data were analysed for each of the 40 training epochs. This served to verify that the training of the respective ANN converged accurately and that possible overfitting issues could be detected. In the case of overfitting, it would be evident that the fitting of the respective ANN shows increasingly improved training accuracies, but the accuracy on the validation data decreases.

Subsequently, the seed for the random training, validation and test data split was varied and the mean and standard deviation over five runs for the different random training and test data sets were determined in order to evaluate the robustness of the training process and the classification results. Thus, error scores were also established, which can be used as rough deviation estimates for the interpretation of the classification results in this paper. The results for different ANN configurations were presented as individual confusion matrices in order to investigate the deviations for each defect category.

Subsequently, the performance of the ANNs were analysed for the synthetic defect images generated by Meister et al. [12] via a DCGAN. The accuracy and loss of the training and validation data were plotted and the confusion matrices for the the results on the test data were calculated. The analyses were carried out only for one single random split of training, validation and test data, assuming that the

error rates determined above also apply as well in a rough approximation. Moreover, only 200 training images, 100 validation images and 100 test images per class were used for the analysis in order to reduce calculation time.
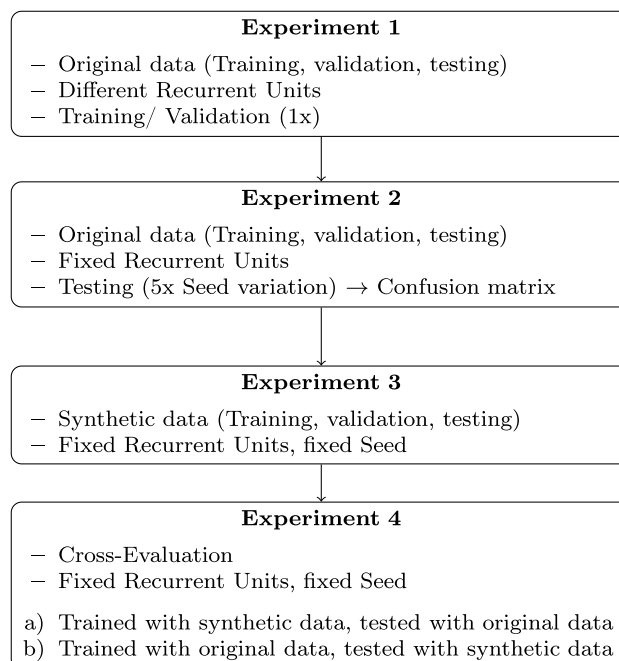
Finally, the validation method GAN-Train GAN-Test [42] already used for fibre placement defect images by Meister et al. [12] was applied to estimate the classification behaviour of the different ANN for altered test data.

The sequence of the different experiments and their characteristical properties are illustrated in the flowchart in Fig. 6 for a better overview.
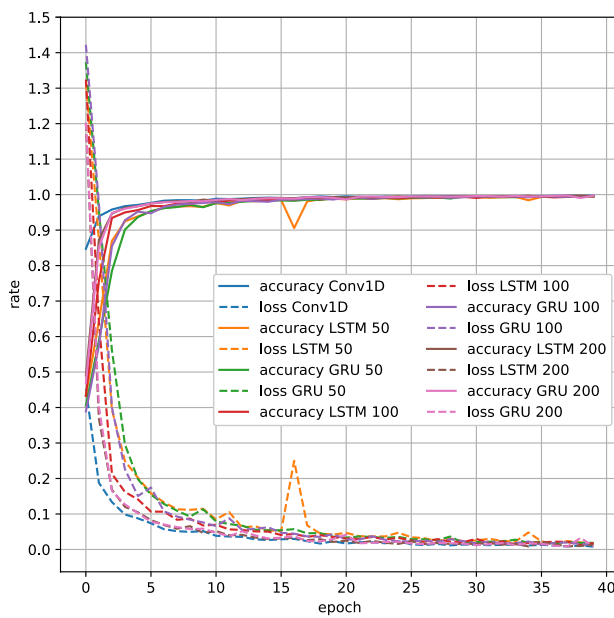
4 Results

4.1 Original process data

Initially, the results for training, validating and testing with real defect image data are discussed. Please note that the data sets are truly different from each other, as described in Sect. 3.4. Figure 7a displays the training history for the training data over the training epochs, with accuracy and loss given on the ordinate for each individual ANN from Sect. 3.3. Accordingly, in Fig. 7b the accuracy is plotted for the validation data per training epoch. Figure 7a shows that during the training process the accuracy converges towards one and the loss towards zero. This can be interpreted as an indicator of a stable training process for the utilised data. In

Experiment 1
- Original data (Training, validation, testing)
- Different Recurrent Units
- Training/ Validation (1x)

Experiment 2
- Original data (Training, validation, testing)
- Fixed Recurrent Units
- Testing (5x Seed variation) → Confusion matrix

Experiment 3
- Synthetic data (Training, validation, testing)
- Fixed Recurrent Units, fixed Seed

Experiment 4
- Cross-Evaluation
- Fixed Recurrent Units, fixed Seed
- a) Trained with synthetic data, tested with original data
- b) Trained with original data, tested with synthetic data

Fig. 6 Flowchart of the sequence of the different conducted experiments

(a) Training history of ANNs having a different number of recurrent units.
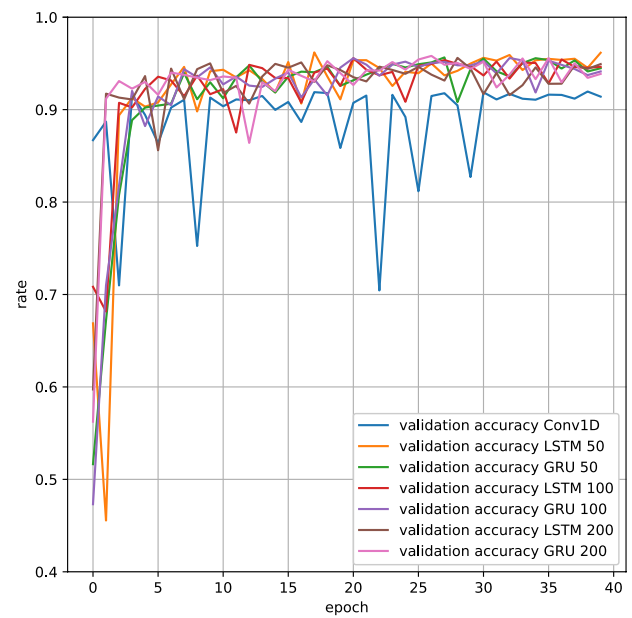
(b) Accuracy of ANNs on validation data set.

Fig. 7 History for training and validation for different ANN architectures

addition, the respective ANN seems to be able to distinguish the training samples well from each other. Figure 7b presents the actual classification accuracy for the validation data set, whereby the results of the LSTMs and the GRUs converge towards a certain value> 0.9 or even continue to rise marginally. The 1D CNN, however, seems to have more problems classifying the validation data correctly at the beginning, but also yield a stable classification level > 0.9 from about epoch 30 onwards. Since none of the configurations reveal a reduction in validation accuracy with increasing number of epochs, it can be assumed that there is no overfitting of the trained ANNs after 40 training epochs, but that a stationary classification state has been reached, which is why 40 training epochs are chosen for the further experiments in this paper.

Figure 8 presents the individual confusion matrices for the classification performance of the trained ANNs on the test dataset. As described in Sect. 3.4, the seed for the random splitting of the data were changed several times and the training and testing steps are repeated. Based on this, mean classification rates and the corresponding standard deviations are calculated.

The overall accuracy is for GRU and LSTM only 2.9–3.7% increased compared to the 1D CNN architecture. Due to the standard deviations of up to 1.84% for certain defect categories, the overall performance is considered to be slightly increased for the RNN architectures but relatively similar across the individual ANN architectures. It is noteworthy that such a high standard deviation is evident across

all ANN architectures for the flawless category. Hence, when looking at the individual defect classes, larger differences are partially evident. None, twists and gaps yield relatively equal classification rate across all ANN architectures. Wrinkles and overlaps show improved classification accuracies for the RNN architectures but lower classification performance for the 1D CNN. Foreign bodies generally have a relatively high misclassification rate compared to the respective overall accuracy. This difference in accuracy is 32.36% for the 1D CNN, 20.85% for LSTM and 26.77% for GRU.

4.2 Synthesised data

In order to assess the classification performance of the developed ANN architectures and configurations for alternative training and test data, the networks are trained and tested with synthetic defect images. Figure 9a presents the loss and accuracy of the training process for different training epochs and Fig. 9b gives the validation accuracy per epoch for the test data. In Fig. 9b it is evident that the training accuracy and training loss converge towards one and towards zero, respectively, and thus a stable training process can be concluded. This behaviour is very similar to the analysis of the real defect images from above. In Fig. 9a the validation accuracy of LSTM and GRU converges towards 0.85. This is slightly less than for the real data and might be related to the design or parametrisation of the upstream convolutional layers in the ANN layouts. This could possibly be related to the construction or parametrisation of the upstream
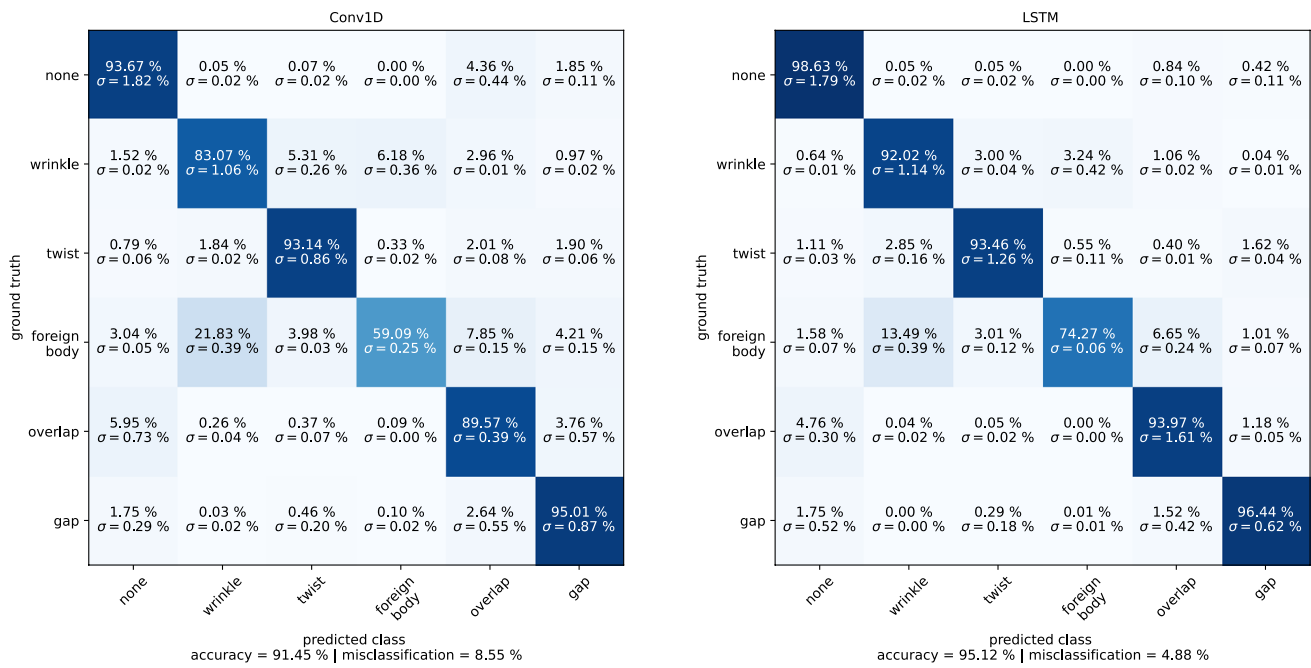
(a) Confusion matrix for the 1D CNN architecture.

(b) Confusion matrix for the LSTM architecture.
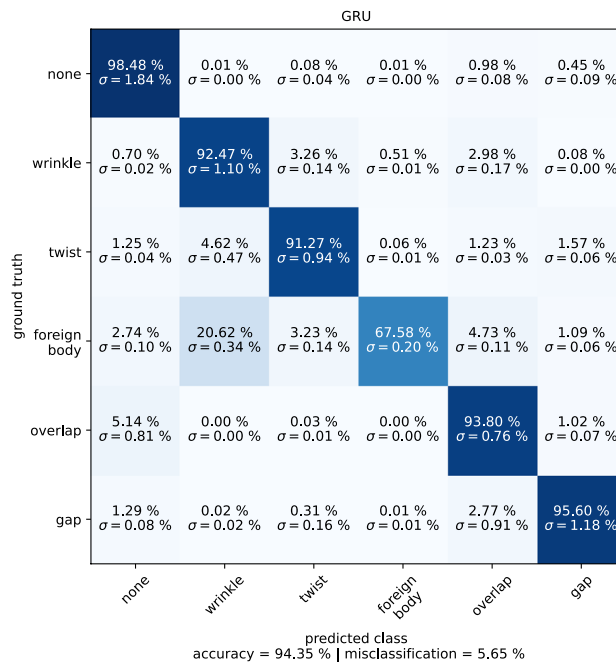

(c) Confusion matrix for the GRU architecture.

Fig. 8 Confusion matrices for the three architectures 1D CNN, LSTM and GRU for different seeds including standard deviation σ tested on the original data

convolutional layers in the ANN layout, as the synthetic data used in this experiment tend to have rounder edges.

The confusion matrices in Fig. 10 show the classification results for the synthetic data in detail. However, only one run was performed to get an impression of the

ANN performance, but to keep the computation time for this much larger data set manageable. Please note that the standard deviation for the synthetic data might be slightly increased compared to the real data since the overall
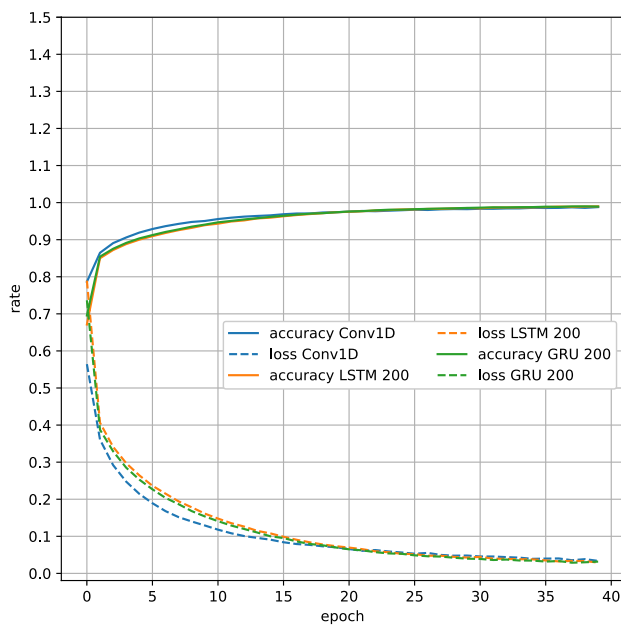
(a) Accuracy and loss for training with synthetic data.
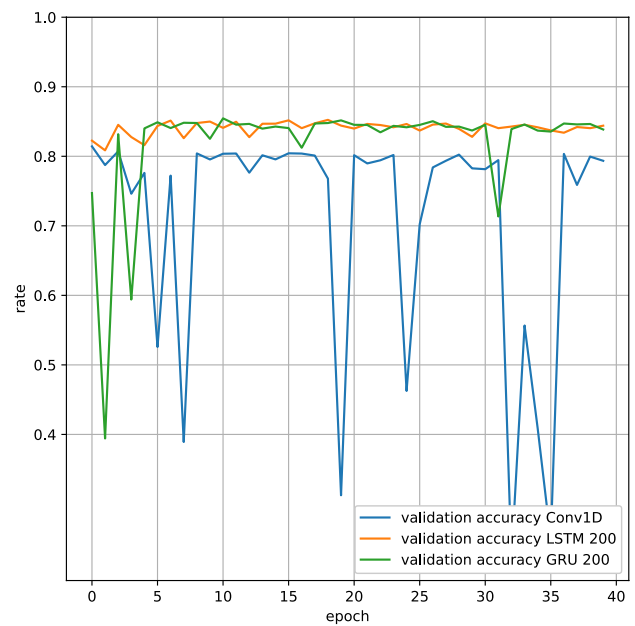
(b) Accuracy for validation on separate synthetic data set.

Fig. 9 The loss and accuracy for the various training processes as well as the accuracy for a different data set are displayed

classification rates are slightly reduced, but it should be in a similar range as for the real defect images.

In general, it is apparent that the overall accuracy for all three ANN layouts is reduced by about 10% compared to the results for the real defect images. Compared to the real defect data, it is observed that almost all classification rates of the individual defects are above the mean classification rates of each ANN. The only exceptions are twists for the LSTM architecture and overlaps across all ANN designs. For twists, the classification rate is only 1.66% below the mean accuracy of the LSTM. Overlaps, on the other hand, have very severe misclassification rates from at least 25.53% for the GRU and up to 34.01% for the 1D CNN. These synthetic overlaps are mistakenly identified as flawless areas within the range of 22.9% and 25.91%, which is critical for practical applications. On the other hand, however, flawless regions are also incorrectly categorised as overlaps by between 7.58% (LSTM) and 13.99% (1D CNN). The misclassification as 'none' is somewhat lower for gaps, but is still between 9.45% and 10.39% for the three ANN setups considered.

Presumably this classification behaviour can be due to artificially generated artefacts within the images, which were added through the DCGAN based synthesis process. Such artefacts are particularly noticeable when comparing synthetic to real data, but do not represent the actual defect. They occur predominantly with less pronounced defect types such as none, gaps and overlaps, as can be seen in the example images in Fig. 4. Thus, the 1D analysis approach

introduced in this paper may have difficulties to classify these defects correctly.

4.3 Cross-validaton

In this final experiment, the performance of the ANNs are evaluated crosswise when trained with synthetic data and validated/ tested with real data, and compared to the performance of the inverse case when trained with real data and validated/ tested with synthetic data. For this purpose, the loss and accuracy for the training processes of the ANNs as well as the validation accuracy for both scenarios are illustrated in Fig. 11. The rates are displayed on the ordinate and the number of training epochs on the abscissa.

At first, it can be seen once more in Fig. 11a, b that the loss and accuracy for the training of all ANN converge towards zero and one, respectively, both for the training with synthetic data and for the training with real data. When training with synthetic data, however, a stationary state is reached after approximately 35–40 epochs, whereas for training with real data this is already achieved after about 20 training epochs. In general, however, a stable training process can be assumed for all considered ANN architectures and both training datasets.

The results are different for the validation accuracy. For training with synthetic data and validating with real data in Fig. 11c, the validation accuracy is around 0.85 for LSTM and GRU, where the accuracy over the various epochs is slightly greater and less fluctuating for LSTM compared to
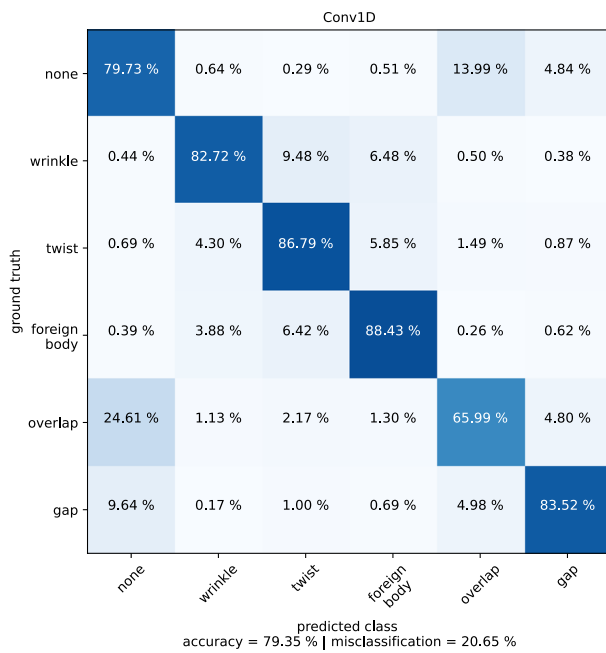
(a) Confusion matrix for the 1D CNN architecture.
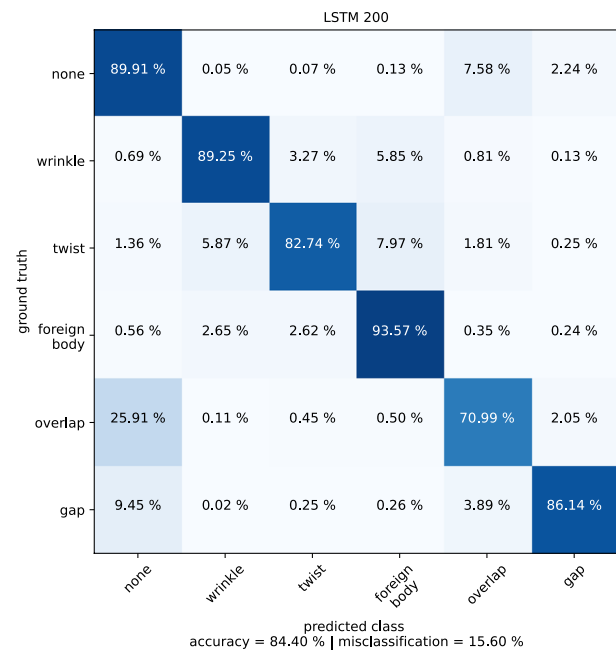
(b) Confusion matrix for the LSTM with 200 recurrent units.
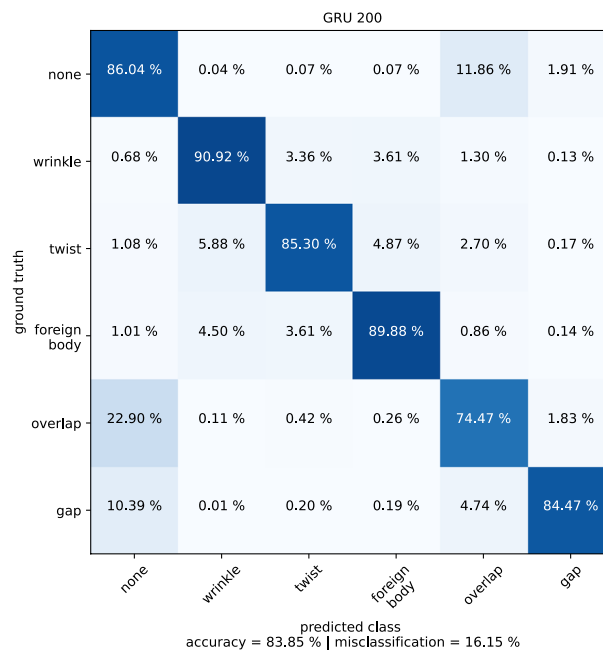
(c) Confusion matrix for the GRU with 200 recurrent units.

Fig. 10 Confusion matrices for the three architectures 1D CNN as well as LSTM and GRU with 200 recurrent units each tested on the synthetic data

GRU. For the 1D CNN architecture, no stable classification accuracy can be observed, which indicates that the 1D CNN has much more difficulty in extracting suitable features from the input defect images.

For training with real data and validation with synthetic data, it is evident again that the validation accuracy of the 1D CNN fluctuates strongly and is very unstable. The validation accuracy of LSTM and GRU stabilises at a relatively low value of slightly above 0.7, which is about 0.15 lower than for the inverse case when trained with synthetic data. From this it can be concluded that the synthetic data represent the real data much better than the other way round.
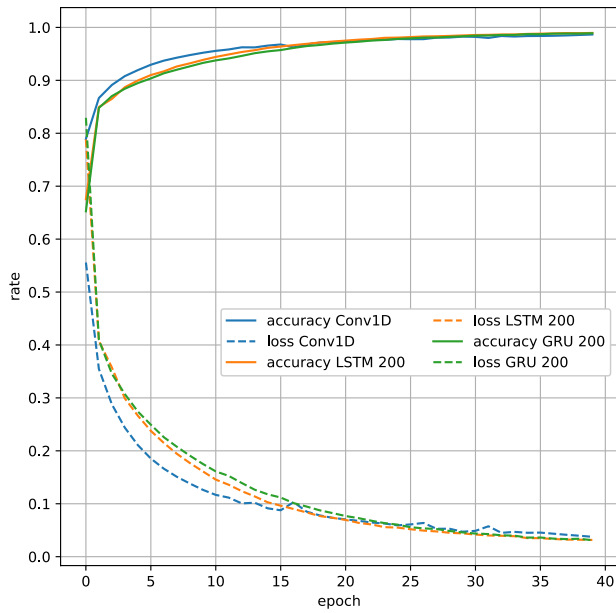
(a) Training history: Trained with Synth., validated with GT
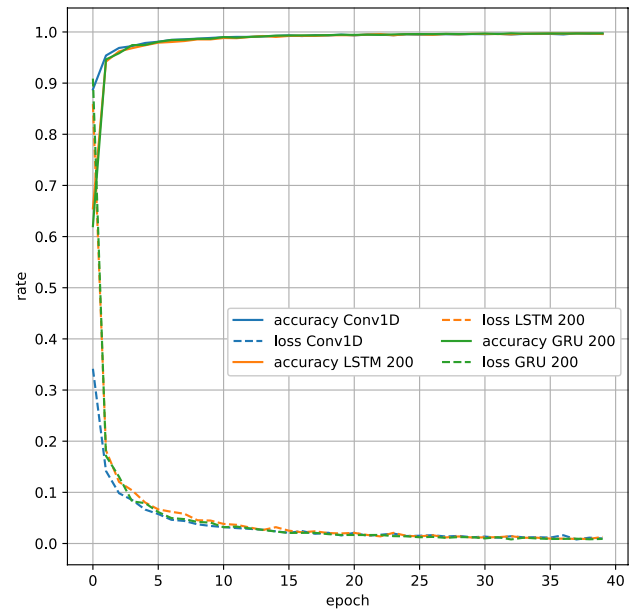
(b) Training history: Trained with GT, validated with Synth.
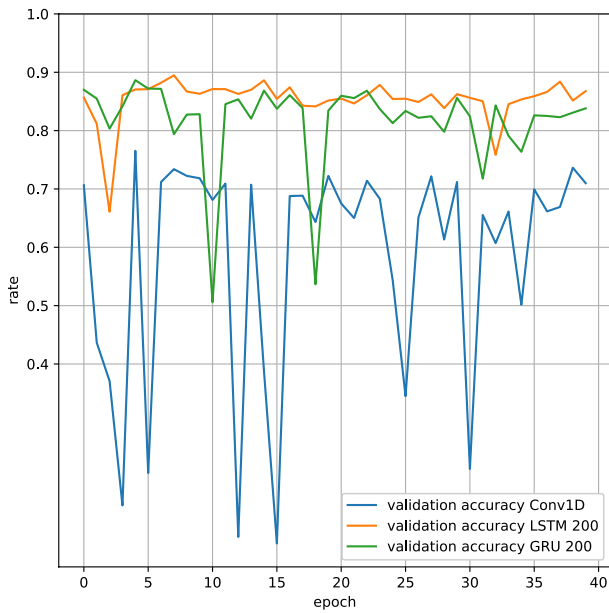
(c) Validation history: Trained with Synth., validated with GT
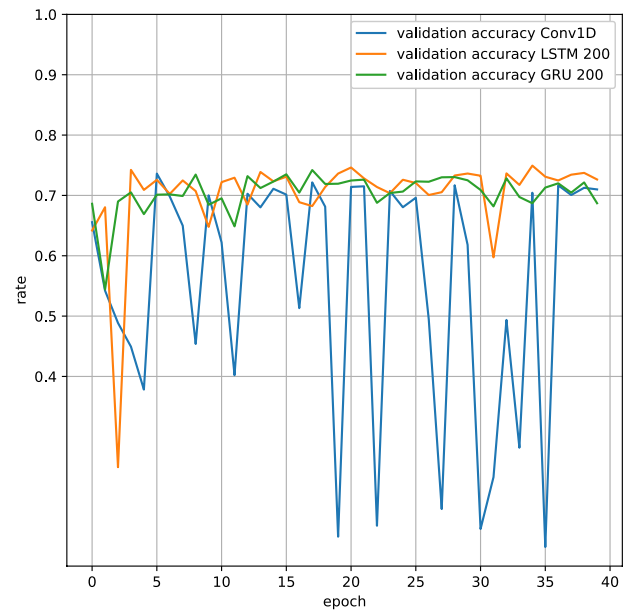
(d) Validation history: Trained with GT, validated with Synth.

Fig. 11 Training and validation histories for different data

However, the significantly lower classification rate compared to the results from Sects. 4.1 and 4.2 suggests that the usage of synthetic data for training ANNs offers minor advantages, especially when sufficient training data for the respective use case is available.

Figure 12 presents the classification results in detail as a confusion matrix, where the matrices on the left side represent the performance for training the ANNs with synthetic data and the right side represents the results for training the ANNs with real data. The best mean classification accuracy
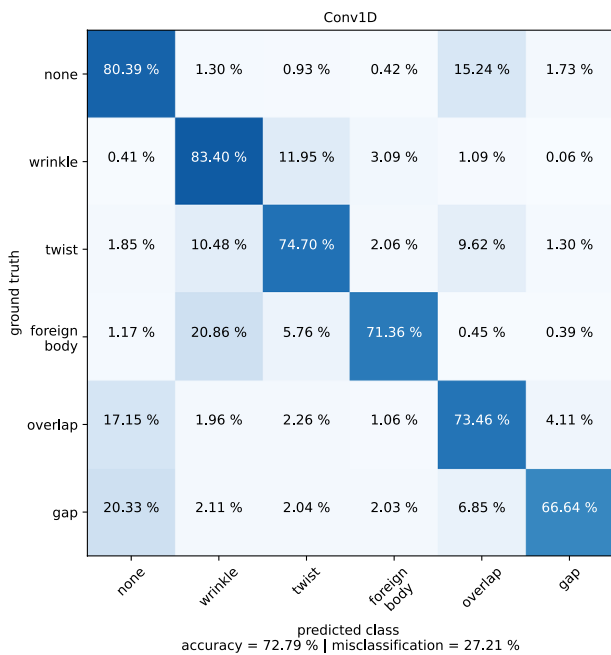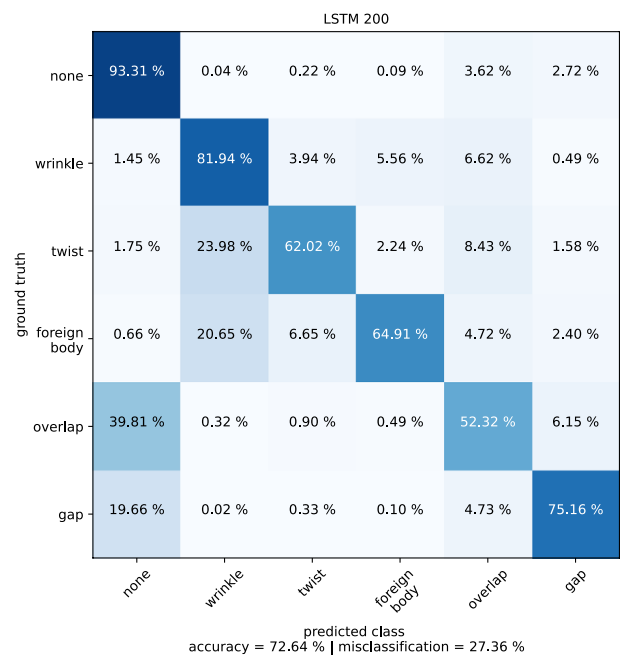
(a) Conv1D trained with Synth., tested with GT
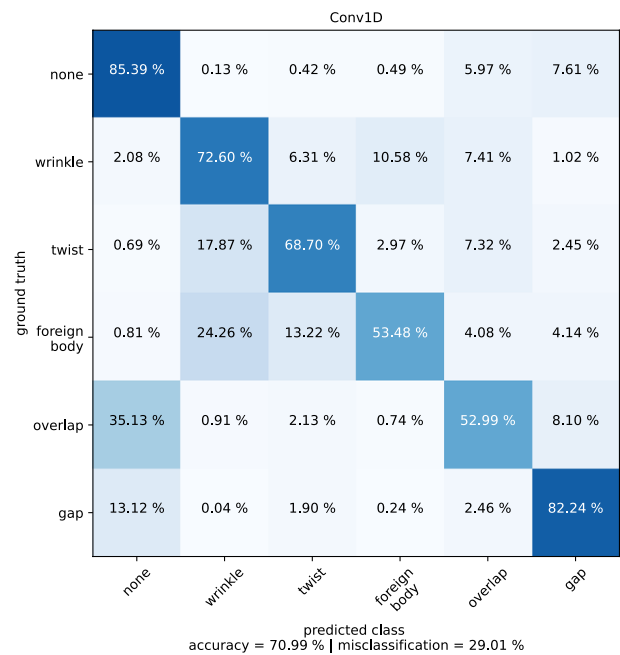
(b) Conv1D trained with GT, tested with Synth.
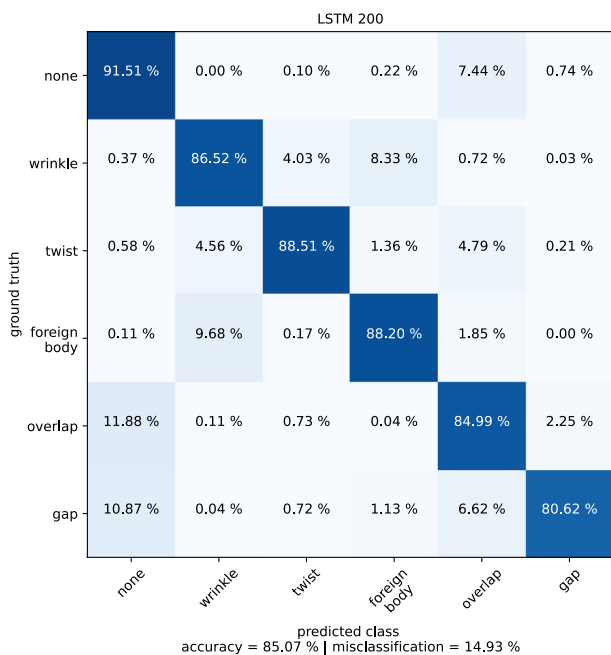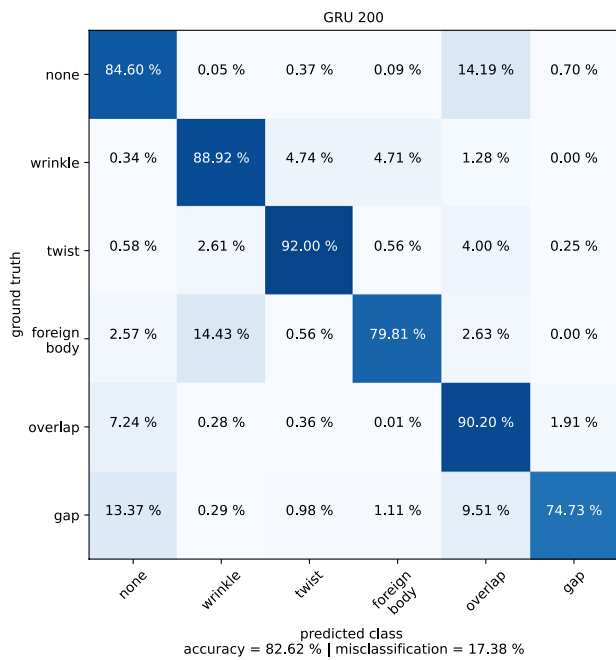
(c) LSTM 200 trained with Synth., tested with GT

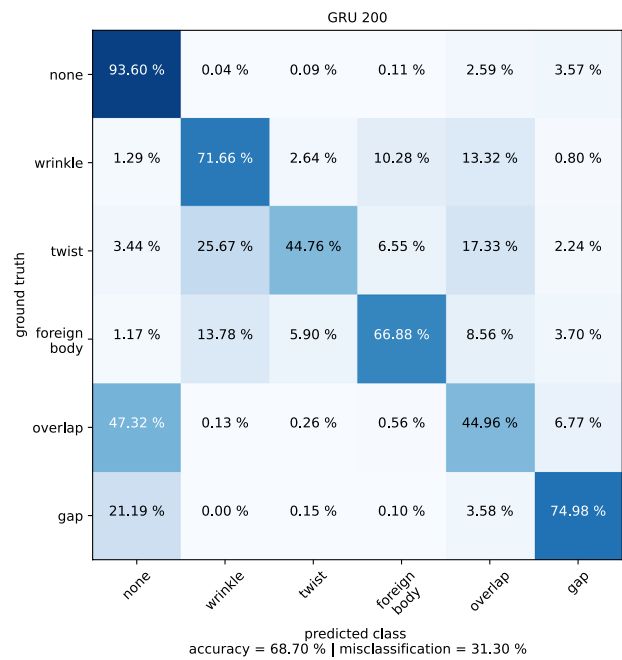(d) LSTM 200 trained with GT, tested with Synth.

Fig. 12 Confusion matrices for various combinations of training and test data

of 85.07% is obtained in Fig. 12c for LSTM with 200 recurrent units, trained with synthetic data. The worst accuracy results in Fig. 12b for the 1D CNN trained with real images.

Furthermore, it is evident that flawless regions yield relatively high classification rates compared to the respective mean overall classification rates with classification scores

(e) GRU 200 trained with Synth., tested with GT

(f) GRU 200 trained with GT, tested with Synth.

Fig. 12 (continued)

between 80.39 and 93.6% across all setups. This also applies to the right column of Fig. 12 where the ANNs have been trained with real data and tested on synthetic data. Moreover, it is noteworthy that gaps are always classified between 4.45 and 7.89% worse compared to the respective mean classification rate when the ANNs are trained with synthetic data, whereas these defects are always classified with at least 2.52% and up to 11.25% greater accuracy in relation to the respective average classification rate when the ANNs are trained with real images. Synthetic gap images do not seem to represent real gap images quite well. Beyond that, an increased tendency of misclassifying overlaps and gaps as none is apparent. For training the ANNs with synthetic data, these misclassification rates are between 7.24 and 20.33%. For training with real data, the misclassification scores are considerably worse with at least 13.12% and up to 47.32%. During training with real data as well as in general for the 1D CNN setup, an increased misclassification of wrinkles, twists and foreign bodies among each other appears. For training the LSTM and GRU with synthetic data, this tendency is significantly lower, except for the misclassification of foreign bodies as wrinkles, which reaches a misclassification rate of up to 14.43%.

The observed lower classification rate with different levels of severity for the individual classes results from the diverging nature of real and synthetic images. This does not necessarily mean that the synthetic data are in general not suitable for the 1D data analysis considered in this study, but artefacts or noise may have been added to the artificial defect images during the data synthesis, which may have a much stronger effect on the 1D analysis case than when considering the entire 2D input image. However, from this it can certainly be concluded that training the ANNs with representative data sets from a given application case is to be preferred, unless a data augmentation is absolutely necessary.

5 Discussion

The presented approach of a line-by-line interpretation of fibre layup defect images provides a sound alternative to the 2D CNN approaches from the previous research of Meister et al. [4, 5, 12]. Especially the issue of having an insufficient amount of training data can be significantly reduced. Using a window-based implementation of the presented method, presumably also defect segmentation like described by Sacco et al. [18, 31] can probably be realised quite well. However, this has not been part of this study. Through the investigations it has been realised, however, that deviations between training and test data or image artefacts potentially have a greater influence on the result of this 1D analysis than it is possibly the case when examining the overall defect image. This concern has also already been discussed in Meister et al. [4]

when examining the input data of a CNN. Accordingly, the results of this paper show that while the amount of image data can be reduced with the presented 1D analysis approach, it is even more crucial to use representative data for training the ANNs.

Regarding the research question, it can be concluded that especially the recurrent network architectures LSTM and GRU with 200 recurrent units show particularly stable classification properties. When representative training and test data are used, average classification accuracies of > 94% can be achieved with these methods. The pure CNN setup, however, reveals quite unstable classification results in certain cases.

The results of this paper are particularly beneficial for developers of LLSS based inline inspection systems, which perform a direct, line-by-line interpretation of the height profile of a surface. The proposed RNN architectures can be trained directly for the respective application case after adapting the input dimension. That enables a rapid transfer of the findings into industrial applications. It can be assumed that the layout and configuration of these ANNs can also be applied beyond inspection in composite manufacturing.

In future research, the presented methodology should be applied to height profiles of an entire fibre placement course with several parallel aligned tows. In this context, a window-based analysis approach or the direct assignment of a classification result to an image coordinate could be investigated. Furthermore, the presented technique needs to be investigated within a realistic manufacturing process on a geometrically more complex component in order to evaluate the method's performance for this use case.

6 Conclusion

Three line-based analysis algorithms were investigated, where LSTM and GRU enable a stable training process with classification accuracies of > 94%. In particular, the ability to perform the ANN training with much smaller datasets is a major advantage of this approach. The classification rate is comparable to that of 2D image analysis methods.

The contribution for the community and the industry are suitable ANN setups, which are suitable for the line-by-line evaluation of topographical data.

Declarations

References

1. Zhang L, Wang X, Pei J, Zhou Y (2020) Review of automated fibre placement and its prospects for advanced composites. J. Mater Sci. 55(17):7121–7155. https://doi.org/10.1007/s10853-019-04090-7
2. McIlhagger A, Archer E, McIlhagger R (2020) Manufacturing processes for composite materials and components for aerospace applications. In: Irving P, Soutis C (eds) Polymer composites in the aerospace industry. Elsevier, New York, pp 59–81. https://doi.org/10.1016/b978-0-08-102679-3.00003-4
3. Eitzinger C (2019) Inline inspection helps accelerate production by up to 50%. Lightweight Design worldwide
4. Meister S, Wermes M, Stüve J, Groves RM (2021) Cross-evaluation of a parallel operating SVM-CNN classifier for reliable internal decision-making processes in composite inspection. J Manuf Syst 60:620–639. https://doi.org/10.1016/j.jmsy.2021.07.022
5. Meister S, Wermes M, Stüve J, Groves RM (2021) Investigations on explainable artificial intelligence methods for the deep learning classification of fibre layup defect in the automated composite manufacturing. Compos Part B Eng 224:109160. https://doi.org/10.1016/j.compositesb.2021.109160
6. Schmidt C, Hocke T, Denkena B (2019) Artificial intelligence for non-destructive testing of CFRP prepreg materials. Prod Eng. https://doi.org/10.1007/s11740-019-00913-3
7. Meister S, Wermes MAM, Stüve J, Groves RM (2021) Review of image segmentation techniques for layup defect detection in the automated fiber placement process. J Intell Manuf. https://doi.org/10.1007/s10845-021-01774-3
8. Huang Y, Cheng Y, Bapna A, Firat O, Chen D, Chen M, Lee H, Ngiam J, Le QV, Wu Y, Chen Z (2019) Gpipe: Efficient training of giant neural networks using pipeline parallelism. In: Wallach H, Larochelle H, Beygelzimer A, d' Alché-Buc F, Fox E, Garnett R (eds) Advances in neural information processing systems 32. Curran Associates, Inc., pp. 103–112. http://papers.nips.cc/paper/8305-gpipe-efficient-training-of-giant-neural-networks-using-pipeline-parallelism.pdf
9. Tan M, Le Q (2019) EfficientNet: rethinking model scaling for convolutional neural networks. In: Chaudhuri K, Salakhutdinov R (eds) Proceedings of the 36th international conference on

machine learning, proceedings of machine learning research, vol 97. PMLR, pp 6105–6114. http://proceedings.mlr.press/v97/tan19a.html

10. Zambal S, Heindl C, Eitzinger C, Scharinger J (2019) End-to-end defect detection in automated fiber placement based on artificially generated data. In: Cudel C, Bazeille S, Verrier N (eds) Fourteenth international conference on quality control by artificial vision. SPIE. https://doi.org/10.1117/12.2521739

11. Meister S, Wermes MAM, Stueve J, Groves RM (2020) Algorithm assessment for layup defect segmentation from laser line scan sensor based image data. In: Zonta D, Huang H (eds) Sensors and smart structures technologies for civil, mechanical, and aerospace systems 2020. SPIE. https://doi.org/10.1117/12.2558434

12. Meister S, Möller N, Stüve J, Groves RM (2021) Synthetic image data augmentation for fibre layup inspection processes: techniques to enhance the data set. J Intell Manuf. https://doi.org/10.1007/s10845-021-01738-7

13. Meister S (2022) Automated defect analysis using optical sensing and explainable artificial intelligence for fibre layup processes in composite manufacturing. Ph.D. thesis, Delft University of Technology. https://doi.org/10.4233/UUID:34442378-E3A2-4C99-865F-57BE3F13B96F

14. Delisle D, Schreiber M, Krombholz C, Stüve J (2018) Production of fiber composite structures by means of cooperating robots. Lightweight Des Worldw 11(2):42–47. https://doi.org/10.1007/s41777-018-0003-2

15. Weimer C, Friedberger A, Helwig A, Heckner S, Buchman C, Engel F (2016) Increasing the productivity of CFRP production processes by robustness and reliability enhancement. In: CAMX 2016-the composites and advanced materials expo and conference. Airbus Group Innovations, 81663 Munich, Germany; AirbusInfactory Solutions GmbH, 81663 Munich, Germany. https://www.researchgate.net/profile/Christian_Weimer/publication/308778487_INCREASING_THE_PRODUCTIVITY_OF_CFRP_PRODUCTION_PROCESSES_BY_ROBUSTNESS_AND_RELIABILITY_ENHANCEMENT/links/57efa78208ae886b8975147a.pdf

16. Black S (2018) Improving composites processing with automated inspection. compositesworld https://www.compositesworld.com/articles/improving-composites-processing-with-automated-inspection. Accessed 19 July 2019

17. Oromiehie E, Prusty BG, Compston P, Rajan G (2019) Automated fibre placement based composite structures: review on the defects, impacts and inspections techniques. Compos Struct 224:110987. https://doi.org/10.1016/j.compstruct.2019.110987

18. Sacco C, Radwan AB, Harik R, Tooren MV (2018) Automated fiber placement defects: automated inspection and characterization. In: SAMPE 18-Long Beach, p. 13. McNAIR Center for Aerospace Innovation and Research, Department of Mechanical Engineering, College of Engineering and Computing, University of South Carolina 1000 Catawba St., Columbia, SC, 29201, USA. https://www.nasampe.org/store/ViewProduct.aspx?ID=11833782

19. Ucan H, Scheller S, Nguyen DC, Nieberl D, Beumler T, Haschenburger A, Meister S, Kappel E, Prussak R, Deden D, Mayer M, Pantelelis N, Zapp P, Hauschild B, Menke N (2019) Automated, quality assured and high volume oriented production of fiber metal laminates (fml) for the next generation of passenger aircraft fuselage shells. In: The Fourth International Symposium on Automated Composites Manufacturing. https://elib.dlr.de/127353/

20. Meister S, Grundhöfer L, Stüve J, Groves RM (2021) Imaging sensor data modelling and evaluation based on optical composite characteristics. Int J Adv Manuf Technol. https://doi.org/10.1007/s00170-021-07591-5

21. Lengsfeld H, Fabris FW, Krämer J, Lacalle J, Altstädt V (2014) Faserverbundwerkstoffe. Hanser Fachbuchverlag. https://www.ebook.de/de/product/22746074/hauke_lengsfeld_felipe_wolff_fabris_johannes_kraemer_javier_lacalle_volker_altstaedt_faserverbundwerkstoffe.html

22. Maass D (2012) Automated dry fiber placement for aerospace composites. In: Composites manufacturing 2012. Danobat

23. Grohmann Y, Stoffers N, Kühn A, Mahrholz T (2016) Development of the direct roving placement technology (DRP). In: ECCM17-17th European conference on composite materials. https://elib.dlr.de/107706/

24. Campbell F (2004) Manufacturing processes for advanced composites. Elsevier Science & Technology https://www.ebook.de/de/product/6827737/manufacturing_processes_for_advanced_composites.html

25. Rudberg T, Nielson J, Henscheid M, Cemenska J (2014) Improving AFP cell performance. SAE Int J Aerosp 7(2):317–321. https://doi.org/10.4271/2014-01-2272

26. Rudberg T (2019) Webinar: Building AFP system to yield extreme availability. CompositesWorld Video

27. Harik R, Saidy C, Williams SJ, Gürdal Z, Grimsley B (2018) Automated fiber placement defect identity cards: cause, anticipation, existence, significance, and progression. In: SAMPE 18 https://www.researchgate.net/publication/326464139_Automated_fiber_placement_defect_identity_cards_cause_anticipation_existence_significance_and_progression

28. Cemenska J, Rudberg T, Henscheid M (2015) Automated in-process inspection system for AFP machines. SAE Int J Aerosp 8(2):303–309. https://doi.org/10.4271/2015-01-2608

29. Black S (2018) Improving composites processing with automated inspection, part II. compositesworld https://www.compositesworld.com/articles/improving-composites-processing-with-automated-inspection-part-ii. Accessed 19 July 2019

30. Gardiner G (2018) Zero-defect manufacturing of composite parts. CompositesWorld https://www.compositesworld.com/blog/post/zero-defect-manufacturing-of-composite-parts. Accessed 18 July 2019

31. Sacco C, Radwan AB, Anderson A, Harik R, Gregory E (2020) Machine learning in composites manufacturing: a case study of automated fiber placement inspection. Compos Struct 250:112514. https://doi.org/10.1016/j.compstruct.2020.112514

32. Zambal S, Heindl C, Eitzinger C (2019) Machine learning for CFRP quality control. In: SAMPE 19. https://doi.org/10.5281/ZENODO.3381930

33. Ronneberger O, Fischer P, Brox T (2015) U-net: convolutional networks for biomedical image segmentation. In: Lecture notes in computer science. Springer International Publishing, pp 234–241. https://doi.org/10.1007/978-3-319-24574-4_28

34. Meister S, Wermes MAM, Stüve J, Groves RM (2021) Explainability of deep learning classifier decisions for optical detection of manufacturing defects in the automated fiber placement process. In: Beyerer J, Heizmann M (eds) SPIE optical metrology-OM106-automated visual inspection and machine vision IV. SPIE. https://doi.org/10.1117/12.2592584. https://spie.org/EOM/conferencedetails/automated-visual-inspection

35. Wu Q, Ding K, Huang B (2018) Approach for fault prognosis using recurrent neural network. J Intell Manuf 31(7):1621–1633. https://doi.org/10.1007/s10845-018-1428-5

36. Lee KB, Kim CO (2018) Recurrent feature-incorporated convolutional neural network for virtual metrology of the chemical mechanical planarization process. J Intell Manuf 31(1):73–86. https://doi.org/10.1007/s10845-018-1437-4

37. Nie X, Xie G (2020) A novel normalized recurrent neural network for fault diagnosis with noisy labels. J Intell Manuf 32(5):1271–1288. https://doi.org/10.1007/s10845-020-01608-8

38. Goodfellow I, Bengio Y, Courville A (2017) Deep learning. The MIT Press. https://www.ebook.de/de/product/26337726/ian_goodfellow_yoshua_bengio_aaron_courville_deep_learning.html

39. Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

40. Cho K, van Merrienboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y (2014) Learning phrase representations using RNN encoder–decoder for statistical machine translation. In: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). Association for Computational Linguistics. https://doi.org/10.3115/v1/d14-1179

41. Heinecke F, Willberg C (2019) Manufacturing-induced imperfections in composite parts manufactured via automated fiber placement. J Compos Sci 3(2):56. https://doi.org/10.3390/jcs3020056

42. Borji A (2019) Pros and cons of GAN evaluation measures. Comput Vis Image Underst 179:41–65. https://doi.org/10.1016/j.cviu.2018.10.009