Fault Classification and Correction based on Convolutional Neural Networks exemplified by laser welding of hairpin windings

Johannes Vater
Planning and Production of Electrified Powertrains
BMW Group
Munich, Germany
Johannes.JV.Vater@bmw.de

Peter Schamberger
Planning and Production of Electrified Powertrains
BMW Group
Munich, Germany
Peter.Schamberger@bmw.de

Alois Knoll
Robotics, Artificial Intelligence and Real-time Systems
Technical Univeristy of Munich
Munich, Germany
knoll@in.tum.de

Daniel Winkle
Planning and Production of Electrified Powertrains
BMW Group
Munich, Germany
Daniel.Winkle@bmw.de

***Abstract*— The automotive industry is facing a change from combustion engine-powered to electrified vehicles. Besides the traction battery, the electric engine is one of the most important components of the electrified powertrain. In order to increase the energy efficiency of the electric motor, wound copper wires are replaced by enameled rectangular copper wires, known as hairpins. In order to produce a conductive connection between hairpins, it is necessary to weld them together. Currently, the automated laser welding of copper is a poorly understood process. Such new production processes are still unknown in comparison to classic engine production and there is only little expert knowledge available. The integration of Industry 4.0 techniques and advanced data analytics provides the opportunity to understand the process of copper welding more thoroughly. A common understanding of advanced data analytics differentiates between predictive and prescriptive analytics. One of the most promising developments in advanced analytics is Machine Learning (ML). There is a wide range of different types of algorithms, theories and methods. An example of these are Convolutional Neural Networks (CNN). They have been designed for learning multidimensional data, such as images or even videos. This paper presents such a CNN to detect welding defects of hairpins. Depending on the classified defect, a rework concept is given (prescriptive analytics). The input parameters are the visual information are derived from of a 3D camera. Using the welding process as an example, the paper illustrates a newly developed method based on the CRoss Industry Standard Process for Data Mining (CRISP-DM) for the development of the CNN. In this context, the paper deals in detail with data preprocessing, modeling and evaluation. The newly developed methodology and architecture of the CNN achieves an accuracy of over 99 percent to predict the defect class.**

***Index Terms*—machine learning, convolutional neural networks, electric motors, hairpin, predictive analytics, prescriptive analytics**

I. Introduction

The electrification of the drive train for electric vehicles requires special requirements compared to standard industrial electric motors in order to be used as an alternative to the combustion engine. These requirements include increasing power density and efficiency combined with a reduction in costs and production time. Therefore, it is necessary to develop new and innovative technologies for the production and to increase the efficiency of electric motors. The so-called hairpin technology is a novel technology to increase the efficiency of an electric motor by replacing the traditional copper windings in the stator of the electric motor with thick copper bars. The main process steps in manufacturing the stator are the deformation of the hairpins, followed by the insertion of the hairpins into the stator's stack of sheets. Afterwards, the free ends of the hairpins are twisted and finally connected via laser welding [1], [2]; however, the problem with this process is that copper has strong reflective properties and can therefore hardly absorb any radiation [3]. For this reason, a higher laser power must be set than for welding steel or aluminum. As a result, the welding of the hairpins can lead to insufficient welding, welding spatter and welding craters. Due to the ever-increasing automation of production facilities, it is essential to detect and eliminate such quality deviations at an early stage. Computer Vision (CV) in combination with ML is an important enabler for this. This allows for the detection of welding defects at an early stage (predictive quality) and for the conception of an inline reworking concept (prescriptive automation) [4].

From this point forward, the paper is structured as follows: Section II provides an overview of the state of the art and the need for action. Section III first introduces the detection of the different failure classes of the welding process and their classification by a CNN. Depending on the fault case, a recommendation for action by a rework concept is derived. This chapter is based on the CRISP-DM method. Finally, Section IV concludes the paper and gives an outlook for future work.

978-1-7281-4319-4/19/$31.00©2019 IEEE

II. State of the Art

A. Industry 4.0 approaches in the field of electric motor production

As described above, new technologies are required to electrify the powertrain successfully. However, this has a far-reaching impact on production technology. In order to face these challenges, a stronger networking of the value chain is essential [5]. Industry 4.0 describes this trend of networking and communication of production facilities in order to optimize the production process [6]. It is determined by a multitude of technologies such as Smart Sensors, Big Data and ML [7].

ML is a kind of artificial intelligence that enables machines or computers to learn a specific task from data. Its aim is to learn this task with the help of data and not through the special programming of rules by experts. The data used for this is called training data. During the so-called training phase, it is used to learn desired rules and regulations. So-called Neuronal Networks (NN) are widespread in this context [8].

As Mayr et al. already described, there are currently only a few ML approaches in the production of electric powertrain [7]. Examples are the use of ML algorithms in thermo and ultrasonic crimping for predictive maintenance, quality management and an ML approach that predicts the cogging torque by analyzing magnet properties, as well as process parameters of a stator [9], [10].

In the field of laser welding in general, there are already many applications in the quality monitoring of laser welding. However, to detect quality deviations with ML algorithms, there are only a few applications common in this area. Furthermore there is only one application of ML methods to detect quality deviations in the welding of hairpins. Thereby the quality of the welding is determined by process parameters as well as with a CCD camera [11].Further approaches of ML algorithms in the production of electric engines are summarized by Mayr et al. [7].

B. Convolutional Neural Network

CNNs, which are an extension of classic NNs, are specifically designed for detecting and classifying features of multidimensional data, such as audio tracks (1D), images (2D), and videos (3D). The typical layer of a CNN consists of three stages. A convolutional stage, a detector stage and a pooling stage The following sections explain the individual stages in more detail [12].

1) Convolution Stage: The convolution stage is the most important component of a CNN. This layer consists of a set of filters, also called convolution kernels. They generate output features through convolution of the input. A filter is a matrix of discrete values that represent the weights of the filters. They are learned during CNN training to extract features [12].

2) Detector Stage: After the convolution stage, the detection stage follows. In this stage, the result of the convolution layer is activated by a nonlinear function, the activation function. This process is very important because it allows a CNN to learn nonlinear connections. The activation function used in this paper is the rectified linear units (ReLU) function [13].

3) Pooling Stage: After the activation is completed, a pooling stage usually follows. In this stage, blocks from the input image of a certain size are combined and summarized into a single value. With Max-Pooling, each block is reduced to the maximum value of this block [13].

4) Regularization techniques: Due to the large number of parameters learned during the training process of an NN, it is possible that the model may over-adapt to the available training data. This can lead to the network learning too highly detailed characteristics of the training data. This causes the network to incorrectly classify data that was not in the training set. This over-adjustment of the parameters during the training process is called overfitting. Drop-Out Layer and Batch Normalization (BN) Layer are two examples of regularization techniques. They can greatly reduce the risk of overfitting [13].

C. Need for Action and Objectives of this Research

As already described in Section II-A, the use of ML algorithms in the manufacture of electric engines in particular is still a little retrograde. Furthermore there is only one application of ML methods to detect quality deviations in the welding of hairpins [11]. As mentioned in Section II-A the authors analyzed the potential of ML for quality monitoring in the laser welding of hairpin windings by using simple 2D images with a low-cost camera. The accuracy for detecting different error classes based on images are in a range from 61 to 91% depending on the error class [11]. However, for industrial applications, this is too low. Therefore, this paper deals with the development of a suitable experimental setup with a 3D scanner, the preprocessing of the data and the appropriate CNN architecture to increase the accuracy. In addition, other defect classes such as weld craters, weld spatters and insufficient welding are considered. Furthermore, a recommendation for action in form of a rework concept is derived depending on the detected defect class.

III. Advanced Data Analytics

A. Experimental Setup

For this application a special experimental setup was designed to generate high-quality and realistic data. Instead of complete hairpins, only thick, already stripped copper wire pieces with a length of $100mm$ are used. This is possible because only the resulting welding cap is important and no complete hairpins are required. The wire pieces are inserted into a test carrier, whereby two pairs of pins can be welded together.

Using Keyence's XR-HT40M 3D camera, it is possible to digitize the welds in the form of 3D data. The advantage of a 3D camera over a classic 2D camera is that height information is used for the inspection process, which improves the stability of the inspection. The disadvantage is the price. In relative terms, a 3D scanner costs 2.5 times more than a standard 2D camera for industrial applications. The recording range of this camera is $16mm$. The measurement resolution in X- and Y- direction is $18.5\ \mu m$. The calculation unit of the camera calculates an RGB image with a special color coding from this 3D height information, which can be output. Fig. 1 schematically shows the procedure for recording the height information and the resulting RGB image of a welding spatter (WS).

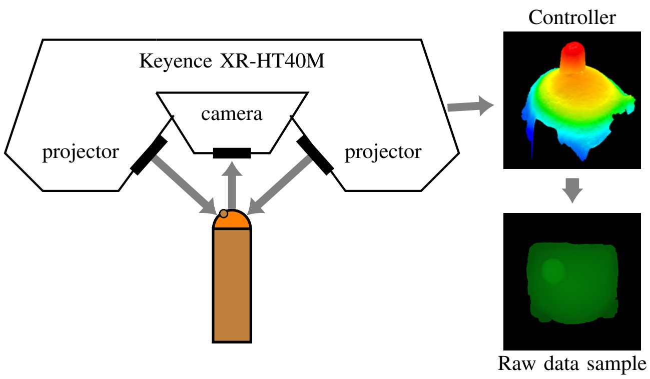


Fig. 1: Experimental setup

B. Business Understanding

Based on the already mentioned unstable laser welding process, faulty welds can arise at the production line of the electrical powertrain. These faults would have a negative impact on the resulting product and therefore it is important to detect these failures inline before they get further processed. The implementation of an automated correction of faulty welds with a suitable rework concept would improve this process enormously.

A technologist analyzed the welding process in detail and defined four important categories of welds that can arise during the welding process in the series production. These classes are divided into four groups, namely correct welding (CW), insufficient welding (IW), welding crater (WC) as well as welding spatter (WS) like shown in Fig. 2. The severity of the considered faults was also varied. This means that correct and insufficient welds of different severity, by varying the laser power, as well as craters and spatters of different sizes were considered. The technologist also verified that the rework concepts are independent of the different severity levels for each class.

Important requirements for the proposed system are high accuracy for the defect detection and also reliable matching for the different rework concepts. Furthermore, an almost real-time capability should be ensured to be able to adapt the production process inline.

C. Data Generation

The amount, the quality, and the class-balance of available data is an important factor for data-driven technologies. Therefore, the procedure of data generation has its goals in generating the largest possible number of data with good quality and under realistic conditions. Based on these conditions, a realistic procedure for generating data with high quality was developed.

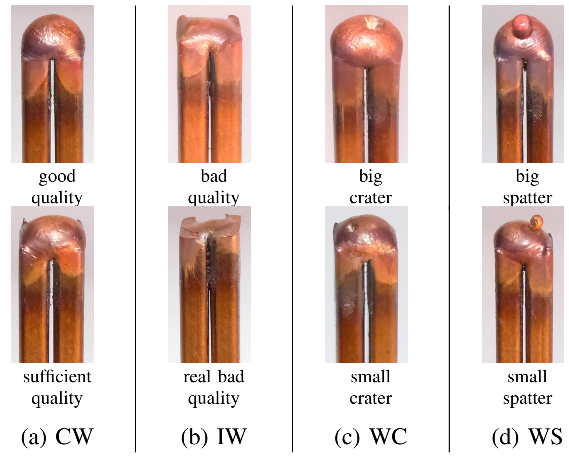


Fig. 2: Representation of the four classes, which result from the welding process of hairpins

As previously explained, failures were produced with different laser-welding programs verified by a technologist under realistic conditions. The resulting welded pins were recorded with a 3D scanner, preprocessed and converted to a grayscale image. As a result of this data generation procedure, it was possible to produce around 550 to 600 labelled images of hairpin welds for each class with different severities.

This data was divided into a training and validation set with a ratio of $0.8/0.2$. To further increase the number of samples in the training set, a data augmentation technique was implemented. By combining rotations, shifts as well as the mirroring of the recorded images, one image could be used to create 49 other similar images. Based on this, more than 90000 training images were generated for the data-driven learning process and about 450 images were available for validating the model. Using the same CNN without data augmentation, the accuracy of the validation set is 93.91 percent. This corresponds to a deterioration of approximately 5 percent, as described in Section III-G. In addition, rotating, shifting, and mirroring images produces realistic data for the training set. In serial production, for example, it cannot be guaranteed that the hairpin image is always centered and without rotation. Data augmentation (rotating and shifting) ensures that the network learns these scenarios as well. In addition, the position of craters and spatters can be synthetically varied by mirroring, allowing the data set to be created without the need for cost-intensive production. This also ensures that the network learns as many cases as possible while training. It is important that the synthetically generated images with the help of data augmentation are not included in the test set under any circumstances. A detailed division of the dataset including data augmentation is shown in TABLE I for an example run.

D. Data Preprocessing

Preprocessing the available data has a huge impact on the resulting performance of a classifier and its accuracy.

TABLE I: DIVISION OF THE DATASET

Class	Training set	Test set	Sum
IW	456	104	560
WS	455	102	557
WC	438	125	563
CW	478	126	604
Sum	1.827	457	2.284
augmented	91350	-	-

Therefore, a specific preprocessing pipeline was developed for getting an optimal input for the CNN classifier and the considered use case. Mainly, this preprocessing pipeline can be divided into the five steps shown in the following. The raw data produced by the 3D scanner is shown in Fig. 1. The color of this image encodes the height information of the welded hairpin. The first step of the preprocessing is calculating the height information of this RGB image produced by the 3D camera. The resultant height information of this welded hairpin is shown in Fig.3.

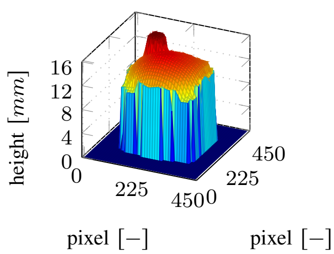
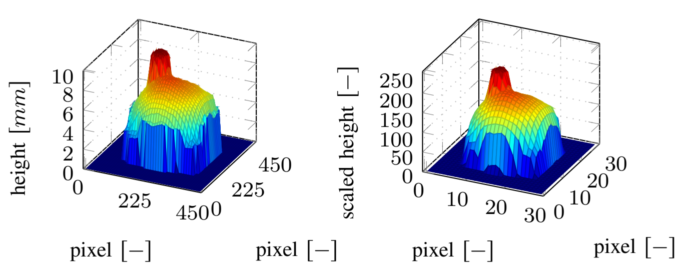


Fig. 3: First step of preprocessing: Hight transformation

After that, the height data is scaled by calculating the median height of the welded pin and defining a specific range above and below this median value. This results in the following height information, as illustrated in Fig. 4a. Next, the x and y dimension of the height data is reduced by applying average pooling with a kernel size of $15x15$ pixels and scaling the height into a range of 0 and 255. This results in two-dimensional data of size $30x30$ pixels, as shown in Fig. 4b.



(a) Second step of preprocessing: Height normalization

(b) Third step of preprocessing: Minimizing

Fig. 4: Preprocessing: Step two and three

In the following step, the three-dimensional height data is interpreted as grayscale image data. The resulting grayscale image, representing the height information with $30x30$ pixels, is visualized in Fig. 5a. Further normalization is an important task to achieve better results with smaller effort for training. Here, a normalization technique based on the subtraction of the mean of one pixel over the whole dataset and dividing it by the standard deviation of these pixels is used. Fig. 5b shows the normalization of a hairpin example with the overall training set.

(a) Fourth step of preprocessing: Grayscaling

(b) Fifth step of preprocessing: Normalization

Fig. 5: Preprocessing: Step four and five

This example visualizes the great benefit of this preprocessing step. The height values of a pixel, which particularly differ from the mean of the same pixel for the whole training set, are highlighted. This especially improves the human as well as the ML capability of classifying the different failures of the hairpin welding task. Summing this up, this normalization technique stabilizes the overall training process and also results in better learning results.

E. Architecture Modeling

The proposed CNN model was basically developed by studying the structure of the well-known VGG networks [8]. This structure is based on convolutional blocks built of multiple convolutional layers, followed by a pooling stage. The further modeling of our structure is based on common suggestions evaluated with experiments [14], [15]. The generic structure of the developed CNN architecture, the input data and the output of the neural network is illustrated in Fig. 6. The elements of the network are described in Section II-B. Therefore, only the structure is described in the following. The model proposed in this paper is made up of four convolutional blocks. The first convolutional block consists of two convolutional layers, each with 8 kernels of size $3x3$ pixels. After this, a Max-Pooling with size of $2x2$ pixels is used. The second, third and fourth convolutional block basically have the same structure as the first one, but with double the number of kernels after each repetition. The second and third block are also followed by a $2x2$ Max-Pooling. The fourth and last convolutional block is followed by a Global-Average-Pooling.

Other important methods used in this model are Batch Normalization before the ReLU-Activation at each layer and performing a Global-Average-Pooling to flatten the output of the last convolutional block. At last there is a Drop-Out layer after the fully connected layer before the final dense layer for each class. These methods were used to guarantee good generalization and to prevent overfitting. The detailed structure of the proposed model, consisting only of about $76,000$ trainable parameters, is shown in Fig. 6. As described in

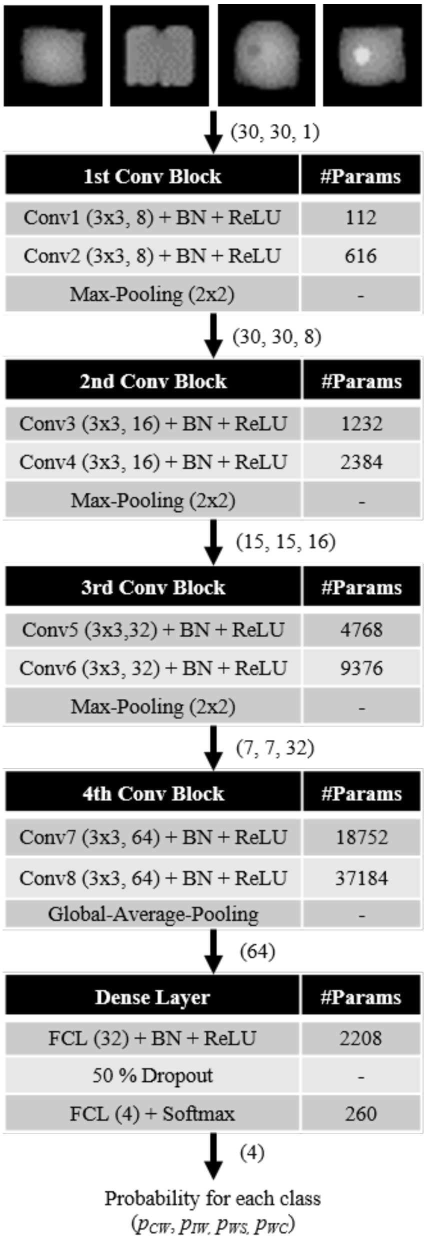


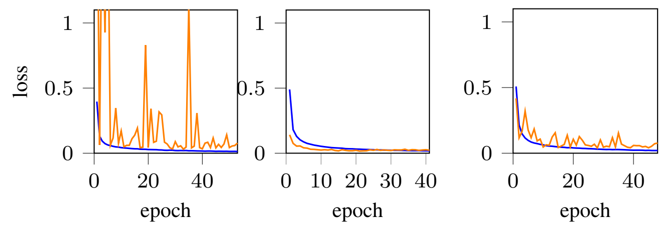
Fig. 6: Structure of the designed CNN

Section III-B, one requirement is an almost real-time capability to detect the class. Therefore, it is important to use a network with as few parameters as possible while maintaining high accuracy.

F. Parameter variation of the model

This section performs a sensitivity analysis of the parameters for the CNN architecture. It varies important parameters of the CNN and examines the resulting accuracy and stability of the training process. These results are compared in graphs, whereby the used parameters of the CNN architecture of Fig. 6 are represented in the middle of the graph. This demonstrates that the proposed architecture has been designed sensibly.

1) Variation of the normalization technique: A factor is the determination of the normalization of the input variables of the CNN. The normalization has a strong impact on the stability of the training process. The resulting loss caused by incorrect or inaccurate classifications when no normalization is used are shown in Fig. 7a. Fig. 7b illustrates the loss curve of CNN training by a normalization dependent on the mean and the standard deviation. Finally, the resulting loss is also examined more closely in Fig. 7c, using a normalization between -1 and $+1$. As Fig 7b shows, by normalizing the pixels based

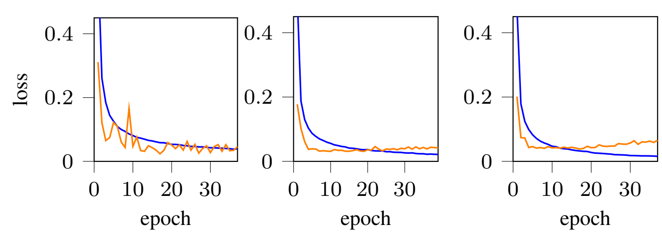


(a) no normalization
(b) Mean and standard normalization
(c) $-1/+1$ normalization

Fig. 7: Resulting curve of the loss of the training data (blue) and the test data (orange) during the CNN training process for different normalization techniques.

on the mean and the standard deviation nearly eliminate the fluctuations. For this reason, this normalization is applied to the training of the final model in order to guarantee the most stable training process possible without fluctuations.

2) Variation of the amount of Conv-Blocks: Another important point for the interpretation of the offered CNN is the depth of the architecture, which in this case is defined by the number of Conv-Blocks. The number of parameters in a CNN is mainly influenced by increasing or decreasing the number of Conv-Blocks. In the following the detailed curve of loss for training for 3, 4 and 5 Conv-Blocks is illustrated in Fig. 8. A detailed analysis regarding the training



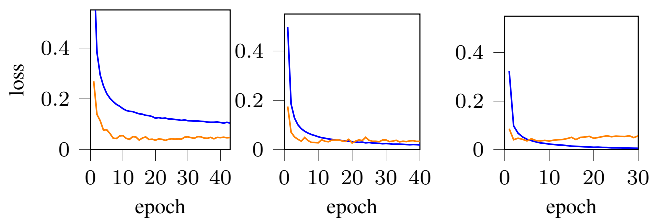
(a) CNN with three Conv-Blocks
(b) CNN with four Conv-Blocks
(c) CNN with five Conv-Blocks

Fig. 8: Resulting curve of loss of the training data (blue) and the test data (orange) during the CNN training process for different amount of Conv-Blocks.

loss for the classification in the Fig. 8a to 8c shows a significant difference. The training process of the modified CNN architecture with 3 Conv-Blocks, as shown in Fig. 8a, shows only small fluctuations. Training this architecture with

4 Conv-Blocks eliminates these fluctuations and results in an almost negligible overfitting. In comparison, training with 5 Conv-Blocks has a much stronger overfitting, which should be avoided, since irrelevant features that are not relevant for the class assignment under consideration are learned in this way. For this reason, the final model was designed with 4 Conv-Blocks.

3) Variation of amount of filters per Conv-Block: Finally, the CNN architecture is analyzed in more detail regarding its width. The width of a CNN, like its depth, has an influence on the number of parameters. For this the number of filters in the Conv-Blocks is halved and doubled and the results are compared. The spelling $4-8-16-32$ means that the first Conv-Block has 4 filters for each convolution, the second Conv-Block has 8 filters and so on. Fig. 9 illustrates the resulting loss curve for the training process by varying the number of filters. By a detailed analysis of the training process



(a) 4-8-16-32 (b) 8-16-32-64 (c) 16-32-64-128

Fig. 9: Resulting curve of loss of the training data (blue) and the test data (orange) during the CNN training process for different amount filters.

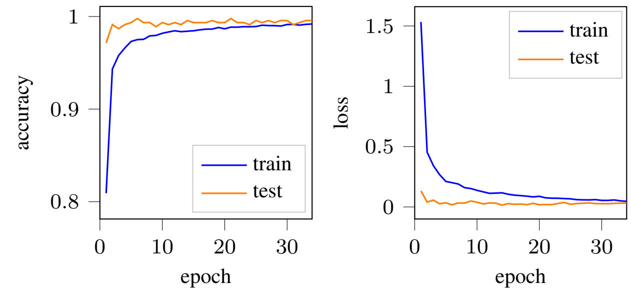
from the Fig. 9a to 9c a conclusion about the number of filters can be made. Fig. 9a shows the large distance between the loss for the classification of the training and validation data. In contrast, in Fig. 9c a small overfitting can be seen, which can only be prevented slightly by early stopping. This overfitting increases by using more filters. Using $8-16-32-64$ filters as shown in Fig. 9b provides the most stable training process.

G. Evaluation

This section first provides a classical validation of the model as shown in Fig. 6 using a classical 5-fold Cross-Validation. Additionally to that, the model is also validated by a special visual method.

1) Classical validation: The proposed model was implemented in keras (version 2.2.4) using the tensorflow backend (version 1.14.0). For training, a stochastic gradient decent optimizer with a learning rate of 1e-4, a decay of 1e-6 and a nesterov momentum of 0.9 was used. For this classification task, a categorical cross-entropy was implemented as a loss-function. The resultant training and validation progress for the model accuracy and loss are shown in Fig. 10a and Fig. 10b. These graphics show that this balanced and smooth training process leads to high accuracy for the training and validation set. Also, these results show that no overfitting occurs.

Furthermore, the method of 5-fold Cross Validation was implemented to reduce the influence of splitting data in the



(a) Resultant training and validation progress for model accuracy (b) Resultant training and validation progress for model loss

Fig. 10: Resultant training (blue) and validation (orange) progress for model accuracy and loss using a 5-fold Cross-Validation

training and test set. This means that there were five trials of training and validating with different combinations for the training and validation set in each run. The resulting confusion matrix for one exemplary run is shown in Fig. 11. In order to

True class \ Predicted class	CW	IW	WS	WC
CW	125	1	0	0
IW	1	103	0	0
WS	0	0	102	0
WC	0	0	0	125

Fig. 11: Confusion matrix of run one

make the results comparable, common rating metrics, such as accuracy, precision, recall and F1-score, were calculated for each run of the 5-fold Cross Validation. These metrics were combined by averaging them for each class in each run. The overall evaluation based on these metrics was calculated by taking the average of all five runs. The resultant accuracy, precision, recall as well as F1-score for each run as well as the averages are shown in TABLE II. These results prove that the proposed CNN-architecture is capable of reliably deciding whether a welding process will result in a failure. A high recall percentage is an especially important requirement for reliable production systems. The proposed model achieves a high average recall of 99.21 percent, which means that on average only 0.79 percent of the failures are unrecognized or classified as a different failure.

2) Visual validation: Further analyzing the decision process of a CNN is an important task. CNNs are mainly black boxes

TABLE II: Accuracy, Precission as well as F1 for each run

Run	Overall Accuracy (%)	Average Precision (%)	Average Recall (%)	Average F1-score (%)
1	$99,56$	$99,56$	$99,56$	$99,56$
2	$99,12$	$99,15$	$99,13$	$99,14$
3	$99,34$	$99,34$	$99,33$	$99,34$
4	$98,68$	$98,77$	$98,68$	$98,72$
5	$99,34$	$99,37$	$99,34$	$99,36$
∅ all runs	$99,21$	$99,24$	$99,21$	$99,22$

without any insights about the structure or behavior for specific decisions. Therefore, special techniques are required to gain insides in order to understand and evaluate the correctness of the decision. A common procedure for analyzing this, called a saliency map, visualizes the importance of pixels for a decision. This is done by setting the color for important pixels to white and black for unimportant ones. Fig. 12 visualizes these saliency maps for different severities of the four classes. The gray images on the left side represent examples of the

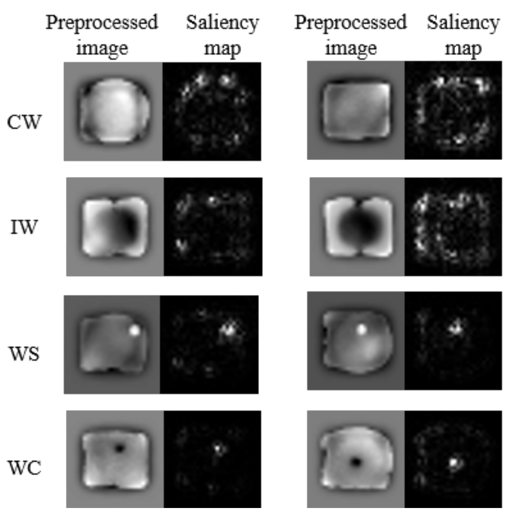


Fig. 12: Saliency map of different classes and severities

preprocessed input to the CNN. The right images are the generated saliency maps for the considered input. These results show that especially for WSs the splatters and for WCs the craters are important features. In contrast to this, the shape is an important feature for CWs and IWs. Based on this result, it can be claimed that the CNN architecture is capable of detecting relevant features for all considered severities and classes.

H. Rework concept

As described above, beside the detection and classification of the failure, additionally a rework concept which is based on the output of the CNN (CW, IW, WC, WS) is given. With the help of a suitable connection between events and the resulting recommendations for action or the adjustments of processes, this procedure can be modeled by a decision tree. This requires special expert knowledge to ensure that the optimal link between action and reaction is established in every situation. Fig. 13 shows the concept for the use case presented in this paper using a decision tree. The blocks thereby represent recommendations for action for the following process. The different blocks are linked by events. These are represented in the first level of the decision tree by the respective welding result (CW, IW, WC, WS). In the following levels, for example, the events are represented by E1. As described in Section III-C, the four main weld classes are CW, IW, WC and WS. They are represented on the first level of the decision tree in Fig. 13. If the result of the welding process is a welding crater, there are two possible options. In Event 1 (E1), the diameter of the crater has a radius greater than $0.8mm$. If this event occurs, it is recommended to remove the hairpin (A). For the second event (E2), the radius of the crater is less than $0.8mm$. As shown in Fig. 13 the decision tree

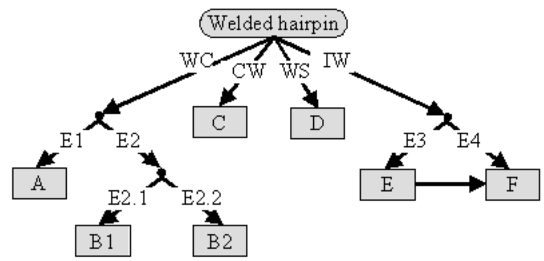


Fig. 13: Decision tree for rework concept

is divided into another level. In Event 2.1, the weld crater is located on the outer part of the hairpin. If this error occurs, reworking with a reduced laser power is recommended (B1). If the welding crater is centered (E2.2), the hairpin should be reworked with an unchanged laser power (B2). If correct welding takes place, the next process step can be initiated (C). If a welding spatter is caused on the hairpin, the spatter must be removed manually by a coworker (D). In the case of insufficient welding, a distinction must be made between two events as shown in Fig. 13. If insufficient welding occurs more than ten times over the last 100 welds (E4), the protective glass of the laser must either be cleaned or replaced (E). The hairpin can then be reworked with full laser power (F). If insufficient welding occurs less than ten times over the last 100 welds (E4), the hairpin can be reworked directly with full laser power.

IV. Conclusion and Outlook

This paper introduces the detection of the welding defects of hairpin windings using a convolutional neural network (predictive analytics). The paper deals with the preprocessing of the 3D data and the modeling of the network. Due to a very good preprocessing and modeling of the network, an average recall of 99.21 percent can be achieved. Depending on the error that occurred, this paper presents a rework concept of hairpin welding (prescriptive analytics); however, a combined IT architecture must be developed, to execute the CNN the model in real-time on the shop-floor. As the authors [4] present in their paper, a combination of edge- and cloud-computing offers the best solution components for the needed requirements. For the use case presented in this paper, it is necessary to analyze, implement, and validate this architecture.

REFERENCES

[1] A. Riedel, M. Masuch, M. Weigelt, T. Glabel, A. Kuhl, S. Reinstein, et al., "Challenges of the Hairpin Technology for Production Techniques," 21st International Conference on Electrical Machines and Systems (ICEMS), pp. 24712476, October 2018.

[2] T. Glaessel, J. Seefried, J. Franke, "Challenges in the manufacturing of hairpin windings and application opportunities of infrared lasers for the contacting process," 7th International Electric Drives Production Conference (EDPC), pp. 17, December 2017.

[3] P. Schmidt, "Laser beam welding of electrical contacts of Lithium-ion batteries for electric- and hybrid-electric vehicles (english) Laserstrahlschweien elektrischer Kontakte von Lithium-Ionen-Batterien in Elektro- und Hybridfahrzeugen (original title)," Doctoral Thesis, Technichal University of Munich, 2015.

[4] J. Vater, L. Harscheidt, A. Knoll, "A reference architecture based on Edge and Cloud Computing for Smart Manufacturing," 28th International Conference on Computer Communications and Networks (ICCCN), 2019, in press.

[5] A. Kampker, K. Kreiskther, M. Hehl, S. Gillen, M. Rothe, "Discrete event simulation approach considering scalable systems and non-expert users in the early phase of production planning for electric powertrains," 6th International Conference on Industrial Technology and Management (ICITM),pp. 139144, 2017.

[6] A. Mayr et al., "Application Scenarios of Artificial Intelligence in Electric Drives Production," Procedia Manufacturing, vol. 24, pp. 4047, 2018.

[7] A. Mayr, M. Weigelt, J. von Lindenfels, J. Seefried, M. Ziegler, N. Urban, et al., "Electric Motor Production 4.0 Application Potentials of Industry 4.0 Technologies in the Manufacturing of Electric Motors," 8th International Electric Drives Production Conference (EDPC), 2018

[8] K. Simonyan, A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," Cornell University, 2014.

[9] A. Mayr, A. Meyer, J. Seefried, M. Weigelt, B. Lutz, D. Sultani, et al., "Potentials of Machine Learning in Electric Drives Production Using the Example of Contacting Processes and Selective Magnet Assembly," 7th International Electric Drives Production Conference (EDPC), 2017

[10] M. Weigelt, A. Mayr, J. Seefried, P. Heisler, J. Franke, "Conceptual design of an intelligent ultrasonic crimping process using machine learning algorithms," 28th Flexible Automation and Intelligent Manufacturing (FAIM2018) Conference, 2018

[11] A. Mayr, B. Lutz, M. Weigelt, T. Glabel, D. Kibkalt, M. Masuch, et al., "Evaluation of Machine Learning for Quality Monitoring of Laser Welding Using the Example of the Contacting of Hairpin Windings," 8th International Electric Drives Production Conference (EDPC), pp. 17, December 2018.

[12] A. Krizhevsky, I. Sutskever, G. E. Hinton, "ImageNet classification with deep convolutional neural networks," Commun. ACM, vol. 60, pp. 8490, 2017.

[13] S. Khan, H. Rahmani, S. A. A. Shah, M. Bennamoun, "A guide to convolutional neural networks for computer vision," Morgan & Claypool Publisher, 2018.

[14] I. Goodfellow, Y. Bengio, A. Courville, "Deep Learning," MIT Press, 2016.

[15] S. H. Hasanpour, M. Rouhani, M. Fayyaz, M. Sabokrou, E. Adeli, "Towards Principled Design of Deep Convolutional Networks: Introducing SimpNet," Cornell University, 2018.