

FAST AUTOMATIZED PARAMETER ADAPTION PROCESS OF CNC MILLING MACHINES UNDER USE OF PERCEPTION BASED ARTIFICIAL INTELLIGENCE

Sebastian Feldmann¹, Michael Schmiedt¹, Johann Jung¹, Julian Marc Schlosser¹,
Tobias Stempfle¹, Christian Rathmann¹, Wolfgang Rimkus¹

¹Hochschule Aalen, Germany

ABSTRACT

This paper concerns unpublished results obtained from the SIMKI (2020) R&D project at the Department of Mechanical Engineering at Aalen University of Applied Science, Germany. The following text generally discusses the development results of the AI-based CNC parameter identification and optimisation tool AICNC. The identification tool supports the AI-based optimisation of milling machine process parameters when using unknown material compositions. The process parameters are determined by a specific test pattern designed to be automatically analysed in real time by a pre-trained perception-based deep learning algorithm. The tool provides the advantage of obtaining real-time quality information due to AI-based quality assessment and the automated identification of material-dependent milling process parameter sets, even for unknown processing material.

KEYWORDS

Artificial Intelligence · Process Optimisation · CNC-Milling · Parameter Prediction · Image Processing

1. INTRODUCTION

The manufacturing sector plays an important role in the gross domestic product (GDP) of the European Union [1]. Nowadays, companies in the metal processing industry are under constant time and cost pressure due to increasing international competition. In addition, documentation and environmental requirements are becoming increasingly stringent [2], [3].

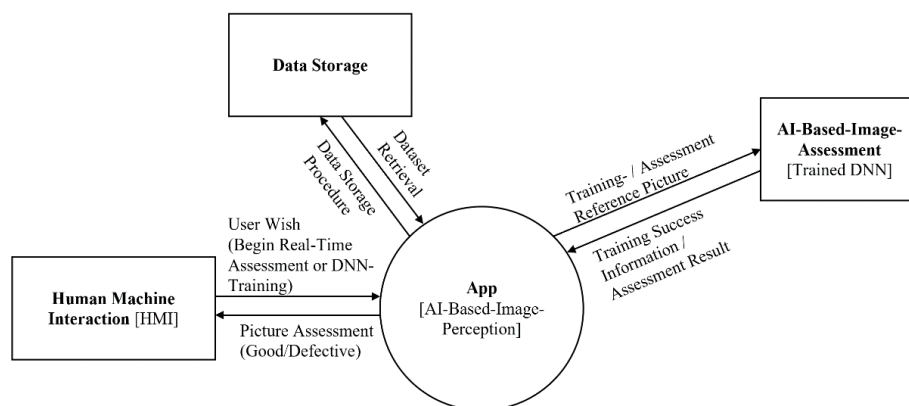


Figure 1. AICNC software module infrastructure overview

In particular, the environmental criteria of reducing material waste and energy consumption in the production process are a significant factor in ensuring the future viability of industrial companies.

In order to increase the efficiency in handling new processing materials with unknown mechanical properties, an AI-based CNC parameter optimisation tool (AICNC) has been developed within the SIMKI research group at the Department of Mechanical Engineering at Aalen University [4], [5]. The software tool uses perception-based deep neural network technology based on the SqueezeNet architecture [6].

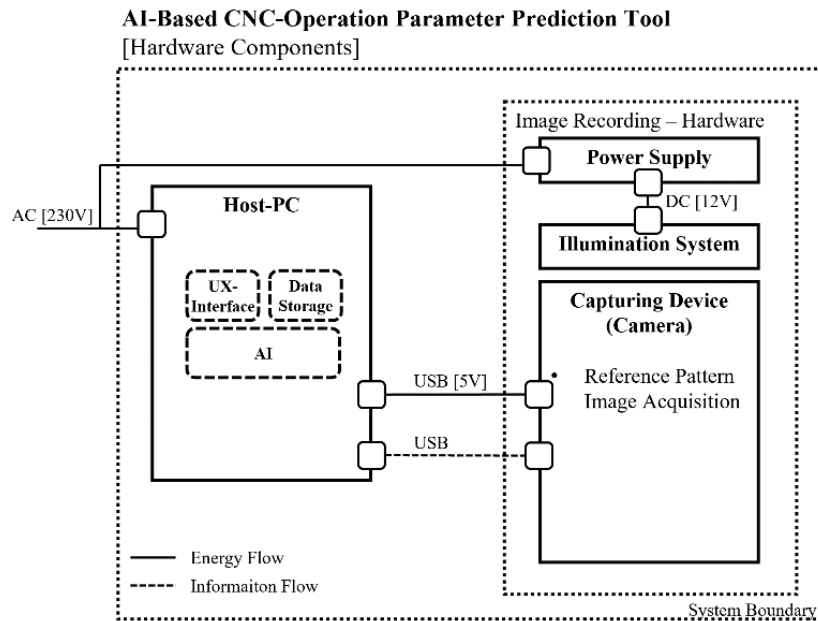


Figure 2. Schematic AICNC hardware interfaces and component representation

The tool features a process to automatically predict the necessary CNC milling parameter settings, e.g. feed and rotation speed of the milling machine, by analysing predefined test patterns. Within the SIMKI research activities, a demonstrator was developed that transfers the theoretical results to an automated software tool that is applicable under industrial conditions [7]. This results in a simple, low-cost solution that significantly increases the efficiency of CNC milling processes and also lowers the environmental impact by reducing material waste.

2. GENERAL ASIC FUNCTIONAL PROPERTIES

Fig. 1 provides a general overview of the AICNC-specific software modules and database components. AICNC basically uses an adapted situationally trained deep learning neural network [DNN], that is trained by specific test patterns of CNC-Milling sample parts, e.g. circles or squares. The DNN is able to identify parts with insufficient surface or edge structures and correlates them to the feed and rotation speed settings applied. The user-interface [HMI] provides 4 general interaction tabs to teach and evaluate the obtained test patterns using the integrated DNN. The application includes a semi-automated image acquisition and snipping tool, which is used to generate training database images under constant conditions. The pretrained DNN can be optimized under use of further images specifically generated within the daily production activities.

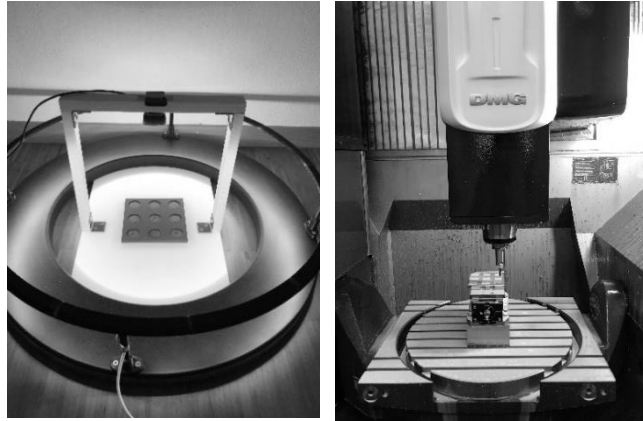


Figure 3. Left: AICNC image acquisition hardware interface setup; right: Test pattern generation on the DMU 65 mono Block 5-axis CNC milling machine reference DMU

The HMI interface provides the ability to capture images, train the network, and classify the best fitting combinations of feed and rotation speed for the unknown material combinations. The final training and pattern assessment results are provided using a layer activation map that indicates the correlated training features of the reference images. Additionally, the layer activation map representation can also be used for real-time inline quality control by an inline image acquisition device. This feature is discussed in further work, e.g. [12], and is not part of this paper.

Table 1. Test pattern milling parameter combination properties of the generic pattern generation for adaptive milling parameter identification and adaption, at the DMU65 milling machine.

Generic Pattern Parameter Set– Curricular Surface						
	Cutting Speed	Feed	Dipping Feed	Cutting - width	Cooling Method	Milling tool
	v_z [m/min]	f_z [mm/min]	f_z [mm/min]	DXY [%]		
1	210	802,2	30	50	Water	HM, coated 10 mm Z3
2	210	401,1	30	50	Water	HM, coated 10 mm Z3
3	210	1604,4	30	50	Water	HM, coated 10 mm Z3
4	420	802,2	30	50	Water	HM, coated 10 mm Z3
5	420	401,1	30	50	Water	HM, coated 10 mm Z3
6	420	1604,4	30	50	Water	HM, coated 10 mm Z3
7	105	802,2	30	50	Water	HM, coated 10 mm Z3
8	105	401,1	30	50	Water	HM, coated 10 mm Z3
9	105	1604,4	30	50	Water	HM, coated 10 mm Z3

Fig. 3 left provides an overview of the actual acquisition infrastructure used during the test period. The module consists of a consumer market video capture device with 720p resolution and a processing rate of 30 frames per second [fps] as well as a field of view [fov] of 68.5°. The hardware is connected to the AICNC software module at a distance of 300mm above the test specimen. To minimise environmental effects and ensure optimal pattern recognition, a circular illumination system (LED) was installed around the capture area, see Fig. 3 left. Fig. 2 represents the corresponding hardware component context and provides an additional overview of the communication and energy interface setup used within the experiment.

The project focused on using widely available hardware components and communication interfaces to minimise investment and integration costs. In subsequent phases, the illumination system can be replaced by software-based compensation for environmental disturbances.

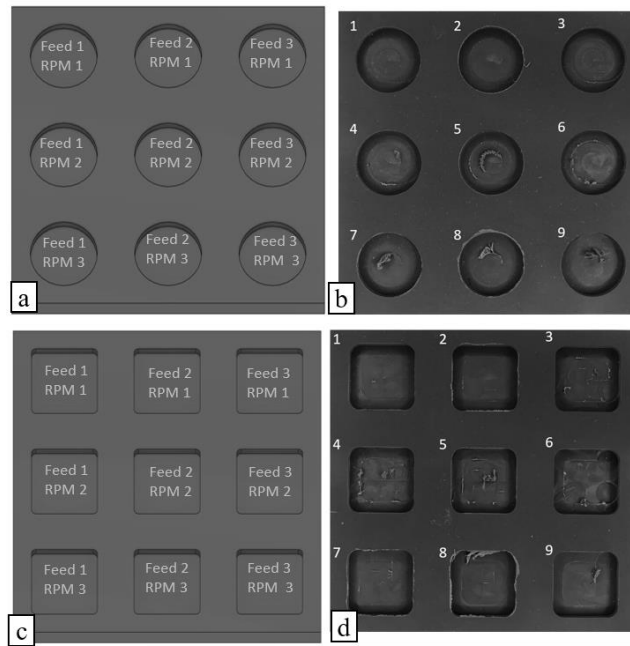


Figure 4. a,c: AICNC test pattern model; b,d: AICNC manufactured test pattern reference sample probe

Figure 3, right, displays a snapshot of the test pattern generation process using the DMU65 milling machine. The test pattern consists of 9 separate areas obtained by specific combinations of machine-related parameter settings, e.g. feed, cutting, etc. As a reference system, the 5-axis CNC machining centre DMU 65 monoBlock was used to apply the procedure under industrial conditions [8]. However, the method can be applied to various machines and processes in this sector. In order to obtain initial machine parameter settings, a literature source (machining manual [9]) was used to specify material dependent initial parameters of the milling machine, e.g. cutting speed $v_c : 210\text{m/min}$ and feed $f_z : 802,2\text{mm/min}$. This parameter set can be safely used to obtain sufficient results but leaves room for increasing the processing speed according to the desired quality criteria. The initial parameter set obtained from literature is used as initial machine parameter set and works as a starting point to apply the generic pattern of AICNC. In order to identify the parameter set in relation to optimum part quality, the test pattern is applied with 8 increasing combinations of cutting speed and milling tool feed.

Table 1 provides an overview of the generic pattern parameter set used in the experiment based on the predefined machine setting in order to identify material dependent processing potential. The identified potential leads to an increase in processing speed and thus to significant time and cost savings compared with the standard parameter settings by manual.

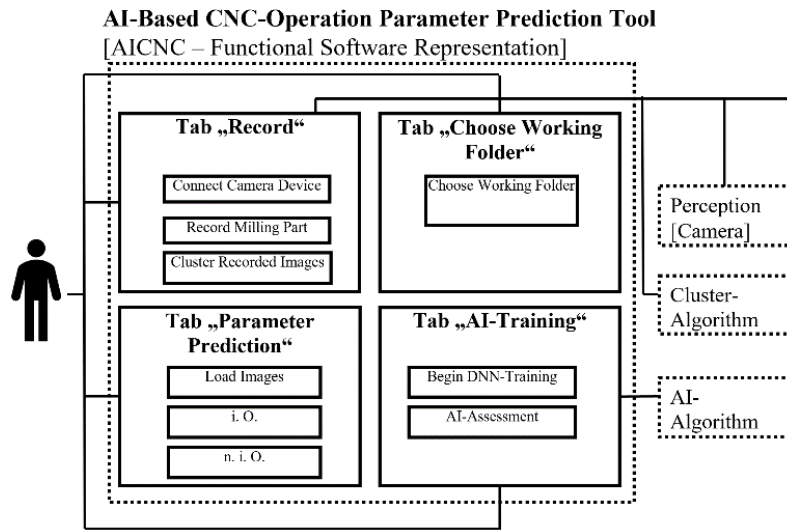


Figure 5. AICNC HMI-interface architecture

Fig. 4. provides an exemplary overview of the generated test patterns and the resulting defect occurrence according to the machine parameter set of Tab. 1. The pattern layers have a size of 140x140 mm and a thickness of 5 mm. The type of material used within the development test procedures was a rigid Polyvinyl Chloride [PVC] synthetic plastic polymer. The typical material parameters are density: 1.3–1.45g/cm³, thermal conductivity: 0.14 to 0.28 W/(mK) and yield strength of 1450–3600 MPa [9]. The generic circular patterns were generated with a diameter of 30 mm and the square patterns with an edge length of 30mm. According to the reference values found in the literature, the machine feed can be increased up to step 3 in order to obtain results of equivalent quality to step 1, compare Fig. 4 – circular pattern. The same result is obtained for the square pattern. According to the square samples in pattern 7, an increased cutting speed also leads to adequate quality results.

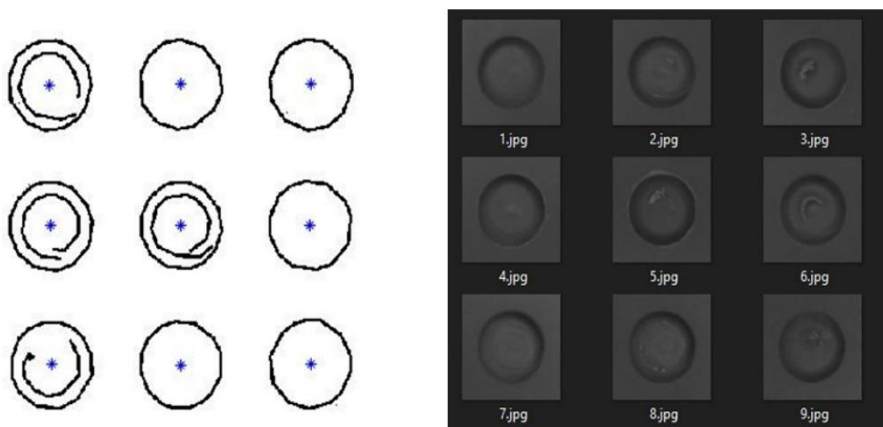


Figure 6. Left: Generic sample pattern edge and centroid detection result right: AICNC segmented test pattern reference sample image

This effect was observed during several test runs. The AI-based parameter prediction tool AICNC automatically clusters the results and stores the samples in a data space according to the good/bad AI-based assessment result. The algorithm is now able to interpolate and provide a

material specific, machine optimised parameter set. Fig 6 represents the pre-processed training pattern image files generated from each test run of the milling machine.

3. AICNC USER INTERFACE AND WORKFLOW REPRESENTATION

The AICNC user interface workflow is generally divided into 4 specific task tabs that guides the user through the necessary steps. The AICNC application communicates directly with the hardware specific image acquisition and clustering methods. Additionally, the DNN represents a separate software instance that is connected to the functional interfaces of the AI training and assessment tabs, compare Fig. 5. In an initial step, the user connects the desired image / video capture device at the 'record' tab. This view allows the user to capture images of the connected device. The “clustering” method performs a segmentation process that includes subroutines for image scaling, edge detection and centroid detection. The captured images of the generic sample patterns are counted according to the centroids found by the algorithm. After the segmentation is finished, the sliced patterns are stored in a common folder. Specific working folders can be selected in the 'working folder' tab. The specified working folder contains the raw image data and the segmented training images.

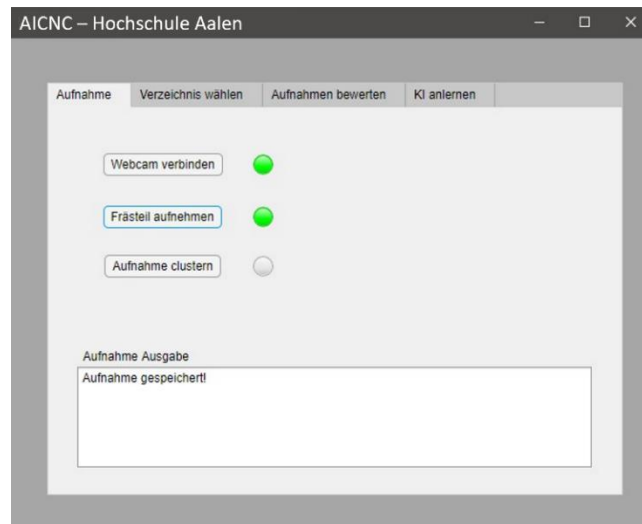


Figure 7. “Record” tab UX-representation

The 'parameter prediction' tab is mainly a user interface that supports the manual classification process of the captured training images. The captured and segmented sample patterns are loaded and analysed by the user. The i.O. and n.i.O button save them to the appropriate DNN training folders. Finally, the “AI-training” tab uses database images to train predefined material patterns. To increase the accuracy, different material types can be used in the DNN pre-training process. The trained DNN on the one hand provides a feedback value as to whether the assessed samples fit the i.O. or the n.i.O. criteria, and on the other hand additionally creates an image of the heat activation layer that marks the important features of the generic sample pattern. The activation map (heat activation layer) is correlated with the backward mapping where the important image features of the damaged parts can be found.

Figure 7 represents the user view in the 'record' tab. Correctly captured and segmented images are indicated by a green-light feedback. Additionally, a continuous system log will be created to validate the user-feedback over time. This is important to correlate differences in the detection

accuracy according to the DNN training process. Fig. 8 represents the AICNC training and assessment tab. This tab provides access to software methods for the DNN training procedure. Once the training process is complete, the system is compatible with a wide range of video capture devices. The assessment button segments the latest captured image and generates a confidential value indicating whether the captured image is i.O. (good) or n.i.O (defective). Additionally, the activation layer display indicates the area and impact of defects according to the intensity colour overlay provided by the user interface.

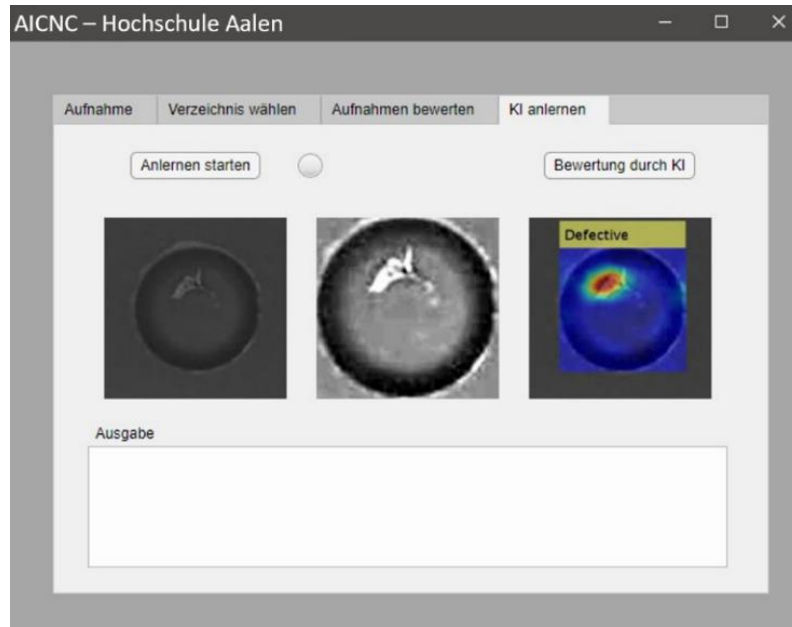


Figure 8. Representation of the AICNC Teaching and Assessment tab

4. DNN TRAINING ENVIRONMENT AND RESULT VALIDATION

The DNN training procedure was performed with 30 'good' rated and 43 'defective' rated images. With 75 training iterations and 150 training epochs. The training PC includes a Core i5-7600K processor with 3.79 GHz clock rate and 16 GB RAM. The operating system based on WIN 10 Pro. The training was performed on a single CPU at a time period of 4:22 minutes to reach a validation accuracy of 100%. The DNN structure is based on the 'SqueezeNet' architecture with exchanged input layers. SqueezeNet is a deep neural network architecture originally developed to reduce the number of parameters and memory size compared to other neural networks, see Tab 2. This architecture consists of several Convolutional Layers, Max Pooling Layers, Fire Layers, a global average Layer, a Fully Connected Layer and an Output Layer.

The input layer supports images as input with resolution of 227x227 and three colour channels, i.e. for RGB images. The image is then convolved with a Convolutional Layer with 64 kernels with the dimensions 3x3. Behind the Convolutional Layers there is a ReLU-activation function and then the output matrices are passed in a max-pooling layer. After that there are 8 so-called Fire Modules. Each of these fire modules consists of a squeeze layer, an expand layer with a filter dimension of 1x1 and an expand layer with a filter dimension of 3x3. The squeeze process consists of a single convolution layer that reduces the number of input channels to a smaller value. This reduces the dimensionality of the input and allows for more efficient processing. The expand process consists of a combination of 1x1 convolution layers and 3x3 convolution layers

that increase the number of channels to a higher level. This combination allows complex features to be captured without requiring too many parameters.

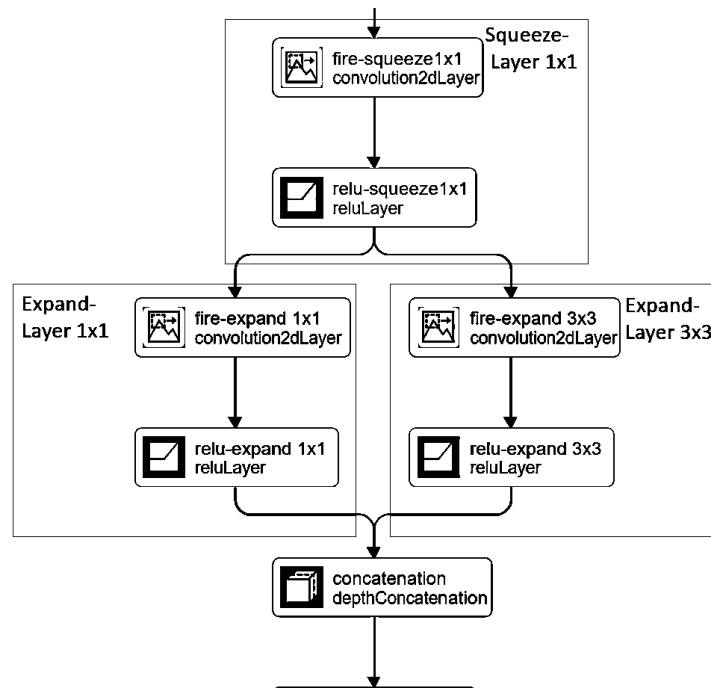


Figure 9. Architecture of the Adapted Fire Module [10]

After these fire modules follows another Convolutional Layer and a ReLU activation function, compare Fig. 9. This is followed by a Global Average Pooling Layer. A fully connected layer was added to the network structure. This consists of two layers, which reduces 1000 parameters to two parameters. From these parameters, the probabilities for the classes "good" and "defective" can be calculated with the help of the Softmax function [10].

Table 2. Architecture of the adapted SqueezeNet DNN [10]

No.	Layer Description
1	Input Layer
2	Convolutional Layer
3	ReLU-Activation-Function
4	Max-Pooling
5	Fire Module 1
6	Fire Module 2
7	Max-Pooling
8	Fire Module 3
9	Fire Module 4
10	Max-Pooling
11	Fire Module 5
12	Fire Module 6
13	Fire Module 7
14	Fire Module 8
15	50% Dropout
16	Convolutional Layer
17	ReLU-Activation- Function

18	Global-Average-Pooling
19	Fully-Connected-Layer
20	Softmax Function

In the presented use case, an ADAM-optimization function was used to optimize the learnable weights and bias parameters. ADAM-Optimiser is an optimisation algorithm for the gradient descent method. ADAM is a further development of the RMSProb and AdaGrad optimizers. However, ADAM also incorporates momentum to improve convergence speed. After calculating the gradient g_t , the moving average of the first and second moments of the gradients is calculated:

$$m_t = \beta_1 * m_{t-1} + (1 - \beta_1) * g_t \quad (1.1)$$

$$v_t = \beta_2 * v_{t-1} + (1 - \beta_2) * g_t^2 \quad (1.2)$$

β_1 and β_2 are the decay factors for the moving average. The default values are used for $\beta_1=0.9$ and $\beta_2=0.999$. Subsequently, this is used to calculate the bias-corrected first moment estimate:

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \quad (1.3)$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t} \quad (1.4)$$

The updated parameters can now be calculated based on the following formula:

$$\theta_t = \theta_{t-1} - \frac{\alpha \hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} \quad (1.5)$$

θ_t is the updated vector of parameters at time t. α corresponds to a fixed step size and ϵ is a small value added to stability to avoid dividing by zero. The parameter update is divided by the adaptive learning rate using the corrected moment estimator and the square root of the corrected second moment estimator. Our SqueezeNet was trained with a step size α of 0.0001. In addition, a learning rate schedule of a Drop factor of 0.7 every 10 epochs was added. An epsilon of 1e-8 was used [11]. The results of the DNN assessment are combined in the 'myNDNet-Postprocess' algorithm and superimposed on the corresponding pattern image. The superimposing process allows one to evaluate the results and identify spots with minor quality. The information is displayed to the user in a separate window, compare Fig. 8. A blue coloration visualizes a good part. The change of colour from green to yellow to red symbolises the increasing degree of damage. Red areas thus indicate defects on the workpiece.

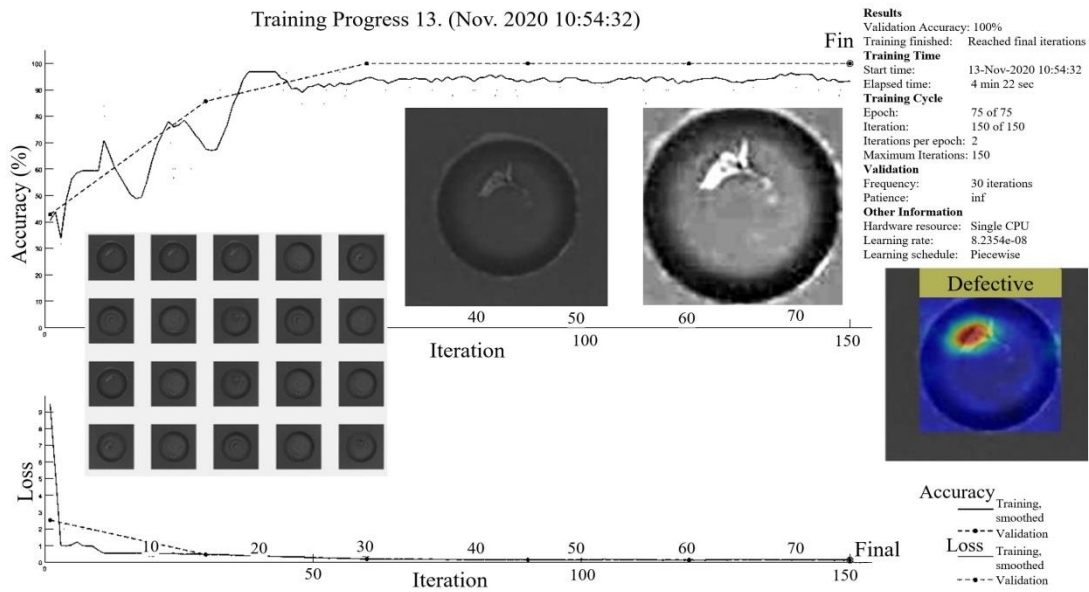


Figure 10. DNN test pattern training progress and resulting defective test pattern layer activation map

This function provides also the possibility to present and compare an image to a previously trained DNN classification result. Thus, the teaching process does not need to be repeated, and validation of the results can be assessed in minimum time. As mentioned, Fig. 8 - right represents the resulting DNN pre-processed test-patterns. At the left area, the acquired sample pattern is visible at the right the originated pattern file. The centre view represents the image processed by the 'myNDNet-Preprocess' function, displayed in the centre view. The third graphic on the right hand, contains related user feedback of the AI classification result. It indicates whether the submitted milling workpiece is i.O. or n.i.O. In addition, the recursive activation layer image colours defective areas identified by the DNN. The dark red area of the activation layer image marks the defective part area. The visualisation of size and position of part defectives supports fast track decision support, whether the part can be reworked or not.

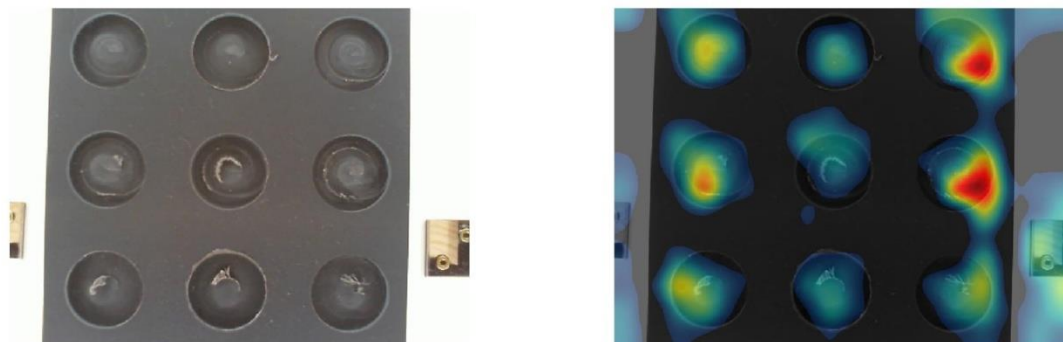


Figure 11. DNN training procedure; left: Milled sample pattern sheet; right: perception based real-time AI defect marking by recursive representation of the DNN layer activation

5. AICNC DNN ASSESSMENT WORK-FLOW

The AICNC work-flow is divided into two separate processes. Initially, the DNN training procedure has to be performed. The training procedure during the development system tests is semi-automatised. The acquired test pattern is automatically cut into partial patterns that fits the

demand of the DNN input network layer, compare Fig. 9. The network structure is divided into the input layer, a hidden layer, and a fully connected layer. Finally, the class output is generated. The application of the used DNN originally targets object recognition tasks. Fig. 6 left displays the reference pattern cuttings, including fully automatised centroid identification. The centroid identification algorithm automatically detects the weighted pattern centre based on the results of the canny edge detector algorithm, compare [12]. Fig. 6 right displays the corresponding pattern cutting images automatically stored in the training folder by AICNC. The training images are named with increasing numbers. The training process demands a cluster of good and a cluster of defective images. With increasing number of training-images, the reliability of the DNN assessment results can be increased within the application of the tool within process usage. Fig 11 . displays the real-time-visualisation overlay of the camera-based perception user-interface. According to testing purposes, the trained DNN was additionally implemented in a consumer-market web-cam device.

At least 9 individual test patterns were needed within the experiment, in order to obtain an initial indication of the desired machine parameter settings. Fig. 11 - left, displays the milled test pattern (camera view) and Fig. 11 - right, represents the real-time camera view including activation map overlay. The camera is generating real-time class-activation map overlay (heat activation layer) with approx. 15-20 fps and indicates under use of phased coloured fields, the occurring extends of part defectives. Areas with high amount of edge and surface defects are marked with a dark red colour and indicate the degree of activation of the DNN layer according to the required features of the part.

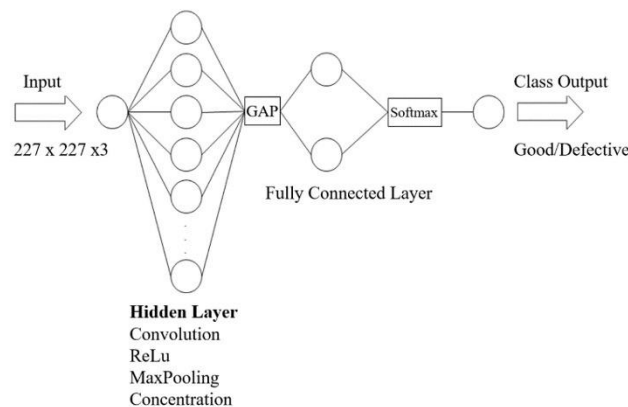


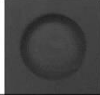

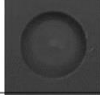

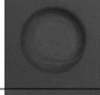




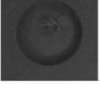
Figure 12. AICNC – DNN layer structure, compare [5]

At this state, the final system is able to provide reliable user-feedback that indicates which areas may provide the optimal milling parameter set. The second column indicates that areas occur with insufficient activation of neurones. This leads to the demand to further harden the DNN with extended training data. Tab. 3 displays the assessed i.O. and n.i.O. sample parts. With the initial samples of 20 patterns a validation accuracy of 100% was reached. Nevertheless, not every sample-pattern was classified correctly. Thus, further experiments must validate the usability of the system. Additionally, an increased database, generated within the industrial system-usage, will help to overcome classification problems. Additionally different DNN-structures will be tested to increase the testing accuracy.

6. CONCLUSIONS AND DISCUSSION

To validate the success of the DNN training at the initial training status, a random seed of 20 images of the category 'defective' was represented and assessed. At the current stage, the AICNC has been initially trained using a small number of test patterns.

Table 3. Initial results of AICNC performance validation

DNN-Assessment Result Validation							
No.	Captured Surface-Pattern	DNN Assessment Result	Expert Assessment Result	No.	Captured Surface-Pattern	DNN Assessment Result	Expert Assessment Result
1		„Good“	„Good“	6		„Good“	„Defective“
2		„No correlation“	„Good“	7		„Defective“	„Defective“
3		„No correlation“	„Good“	8		„Defective“	„Defective“
4		„Good“	„Good“	9		„Good“	„Defective“
5		„Good“	„Good“	10		„Defective“	„Defective“

The test results from the perception-based real-time capturing device provide sufficient results to support milling parameter identification under industrial conditions. In future development steps, the AICNC will be trained under industrial conditions with an extended database. Furthermore, the network structure will be replaced by the VGG-19 [13], a more specialised DNN structure. This network uses 19 layers and is also mainly used for object recognition. Internal tests have already shown better results. That means that key users will use the software in customer projects.

ACKNOWLEDGEMENTS

The presented procedure results from research activities within the SimKI-Project 'Echtzeitdatenerfassung und Parameterkorrektur mittels einer mit Simulationsdaten angelernten KI'. The project was managed by Dr. Wolfgang Rimkus and Prof. Dr. Sebastian Feldmann. Funding was provided by the Ministry for Economy, Work and Housing in Baden-Württemberg (MFW) as part of the research programme 'KI Innovations Competition' in the year 2019.

REFERENCES

- [1] Sáenz de Argandona E., Aztiria A., García C., AranaN., Izaguirre A., Fillatreau P. 'Forming processes control by means of artificial intelligence techniques'. Journal of Robotics and Computer-Integrated Manufacturing, 2008, doi: 10.1016/j.rcim.2008.03.014.
- [2] Europäische Kommission, 'Mitteilung der Kommission an das Europäische Parlament, den Europäischen Rat, den Europäischen Wirtschafts- und Sozialausschuss und den Ausschuss der Regionen: Green Deal der Europäischen Union', Brüssel, 2019

- [3] Romano P., 'European Manufacture of the Future, role of research and education for E European leadership', MANUFUTURE 2003 Conference, Italy, 2003.
- [4] Schlosser J. M., Schneider R., Rimkus W., Kelsch R., Gerstner F., Harrison D. K., Grant R. J., 'Material and simulation modelling of a crash beam performance – a comparison study showing the potential for weight savings using warm-formed ultra-high strength aluminium alloys', Journal of Physics, 2017, Vol. 896, pp. 12091, DOI 10.1088/17426596/896/1/012091
- [5] Feldmann S., Kempter G., Esslinger R., Tran H. T., 'Support of Image-Based Quality Assessment in Discrete Production Scenarios through AI-Based Decision Support', 13th International Conference on Advancements in Computational Sciences, ICACTE, 2020, ISBN: 978-1-4503-7732-4 6. Iandola F. N., Han S., Moskewicz M. W., Ashraf K., Dally W. J., Keutzer K., 'SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and 0.5MB model size, 2016, arXiv:1602.07
- [6] [Online]. Available at: <https://www.wirtschaft-digital-bw.de/ki-made-in-bw/innovationswettbewerb-ki-fuer-kmu/simki-echtzeitdatenerfassung-und-parameterkorrektur>, 2020
- [7] [Online] x-technik IT und Medien GmbH, 'DMU 65 monoBLOCK CNC universal milling machine'. [Online]. Available at: https://www.zerspanungstechnik.com/bericht/horizontal-bearbeitungszentren/dmu-65-h-monoblock-mit-bearbeitungszentrum-maximal-flexibel-in-der-massenproduktion_2022-01-01, accessed in November 2020.
- [8] Hoffmann Group, Fraunhofer-Institut für Werkzeugmaschinen und Umformtechnik, 'Zerspanungshandbuch: Bohren, Gewinde, Senken, Reiben, Sägen, Fräsen, Drehen, Spanen, Präzisions-schleifen', 2. Aufl. München: Hoffmann Group, 2014, ISBN: 3-00-016882-6
- [9] Iandola, Forrest N., et al. "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <math>< 0.5\text{MB}</math> model size." *arXiv preprint arXiv:1602.07360* (2016).
- [10] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimisation." *arXiv preprint arXiv:1412.6980* (2014).
- [11] Rong W., Li Z., Zhang W., Sun L., An improved Canny edge detection algorithm, IEEE International Conference on Mechatronics and Automation, 2016, doi: 10.1109/ICMA.2014.6885761
- [12] Russakovsky, O., Deng, J., Su, H., et al. 'ImageNet LargeScale Visual Recognition Challenge', International Journal of Computer Vision (IJCV). Vol 115, Issue 3, 2015, arXiv:1409.0575v3
- [13] Feldmann, M. Schmiedt, J. M. Schlosser, W. Rimkus, T. Stempfle, C. Rathmann, 'Recursive quality optimisation of a smart forming tool under the use of perception-based hybrid datasets for training of a Deep Neural Network', Journal of Discover Artificial Intelligence, 2022, doi: 10.1007/s44163-022-00034-4.