KI Based Chip Classification for Detection of Unwanted Chip Buildup

Joshua Hermann, Roman Radtke, Aibek Kabanbayev, Junisbekov Mukhtar Shardarbekovich, and Alexander Jesser

***Abstract*—When Materials are machined with subtractive manufacturing the material will be chipped resulting in scrapings of the base material. Those scrapings can get tangled up with the tool, leading to damages of the part and-or the tool. Currently, the machining will continue until the assigned worker notices the chip buildup. Then there already could be scrapings and damaged parts must be machined again. To prevent this issue, a mounted camera capture images of the machine tool. These pictures will be classified by an Image Classification model differentiating between tangled up and harmless scraps. A Residual Convolutional Neural Network was trained to learn the difference between the two classes on a very limited labeled dataset. When a harmful type of chip is recognized by the Image Classification model, preprogrammed measures can be initiated e.g., stopping the process, changing the tool, or changing the process parameters. This should contribute to the possibility to get the maximum service life from each tool without the need of highly specialized personal supervision of the whole machining process while also minimizing the chance of tool breakage.**

***Index Terms*—Image classification, convolutional neural network, residual networks, data augmentation.**

I. Introduction

Depending on the material and the machining process, chips are produced during the subtractive machining of a work piece. These chips can have a negative influence on the process, ranging from slight damage to the surface to the production of a scrap part due to tool breakage.

By analyzing the chips produced by an optical system an undesirable chip development (Fig. 1) can be detected and a corresponding reaction by the user of the machine or the machine itself can be initiated to break the chip or to replace the tool in use. Possible machine reactions could be to change the cutting speed or a brief reversal of the direction of rotation of the tool to break the chip or similar. Best case is the development of a short chip that breaks on the tool itself leading to a short chip that falls of and is washed away by the coolant.

This type of chip is called short chip and is desirable. Less favorable is a chip that does not break and grows on the tool: the so-called long chip. Short chips rarely cause scratches on the surface of the work piece. A long chip, on the other hand, often scratches the surface of the material due to the chip turning along the tool - in addition, the chip can wrap around the tool, which also has a negative effect. In extreme cases, the resulting chip can "clog" the tool, or the resulting forces become so great due to the coiled chip that the tool can break.

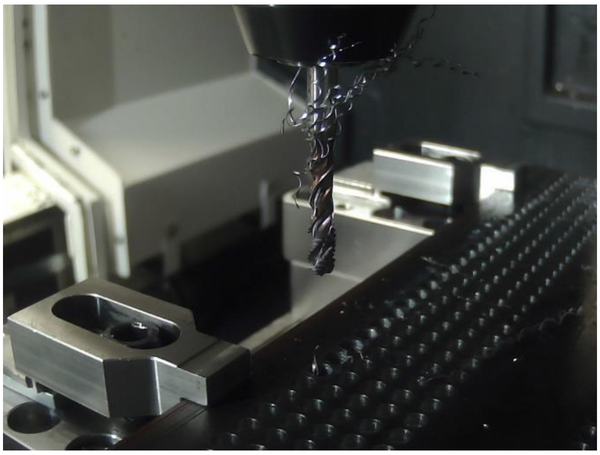

Fig. 1. Chips coiling around the tool might result in tool failure.

II. Related Work

The topic of failure detection has been researched in different fields like 3D printing [2][1] Medicine [2] or semiconductor production [3]. But so far there has been no failure detection for metal machining. In all that research, CNNs with trained data has been used to classify failures. But in contrast to our issue, usually the networks are trained to detect failures after the part is machined, so the material is already wasted. In our case, we detect the symptom (coiled chips) to stop any material waste before it happens.

III. Object Detection and Object Classification

To solve the problem of chip classification first it had to be determined if an Object Detection Algorithm or an Object Classification Algorithm is the method of choice.

Object detection uses a Neural Network searching for all objects in an image and outputs their position data. Since in this project the installed camera is always in the same position relative to the tool (Fig. 2), the use of object detection is unnecessary. In addition, object detection is associated with a higher effort since the positions must also be determined when processing the training data [4].

With Object Classification the Neural Network analyses an image and only outputs what type of object is located on the image which is sufficient for the set goal.

Manuscript received July 6, 2022; revised August 19, 2022.

Joshua Hermann, Roman Radtke, and Alexander Jesser are with University of Applied Sciences Heilbronn, Germany (e-mail: joherman@stud.hs-heilbronn.de, roman.radtke@hs-heilbronn.de, alexander.jesser@hs-heilbronn.de, alexander.jesser@hs-heilbronn.de).

Aibek Kabanbayev and Junisbekov Mukhtar Shardarbekovich are with t Dulaty University Taraz, Kazakhstan, (e-mail: aibek.kabanbayev@hs-heilbronn.de, d_muhtar@mail.ru).

doi: 10.18178/ijfcc.2022.11.4.592

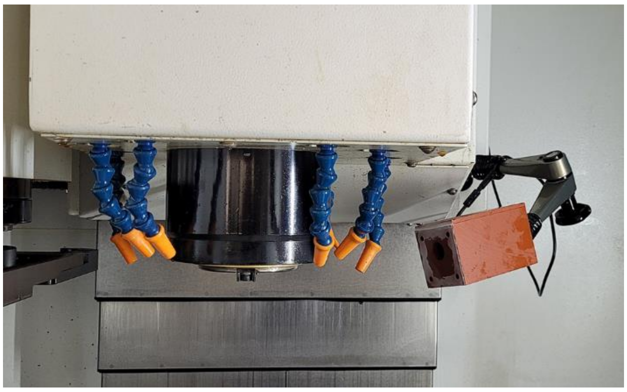

Fig. 2. The camera is mounted in a fixed place in the machine related to the tool position hence Object Detection is obsolete.

IV. CNN Architecture and Classification

The basic type of a Neural Network is the "Convolutional Neural Network" (CNN). Convolution describes the process of parsing an image with a constant sized section (e.g. 3x3 pixels), and evaluating each of these sections. While in signal processing a predefined function is often used to evaluate these image sections in CNN it is this function that is learned during training of the network in a prior step. It is also important to note that convolution reduces the size of the image depending on the core size. This structural change must be considered when creating a CNN with multiple layers (see Fig. 3).

Consequently, after the convolution operation, a map of the image is obtained. In this map all important areas that could contain the searched object are marked. Now the areas which were evaluated best are selected and passed on to another non-convolutional Neural Network [5].

This Neural Network will then output the classification. To do so it is important that the last layer is exactly as large as the number of object classes. In our project we want to detect chips building up around the tool as a worst-case-scenario and hence have two classes: chip winding or no chip winding. Therefore, the last layer of our network has 2 parameters which also resemble the output of the classification. These values are finally normalized using a *softmax* function. Thus, the sum of all values is always 1 and we can see the confidence for each parameter respectively each class. If for example the first value is 0.25 and the second value is 0.75 the confidence value for no chip buildup is 75%.

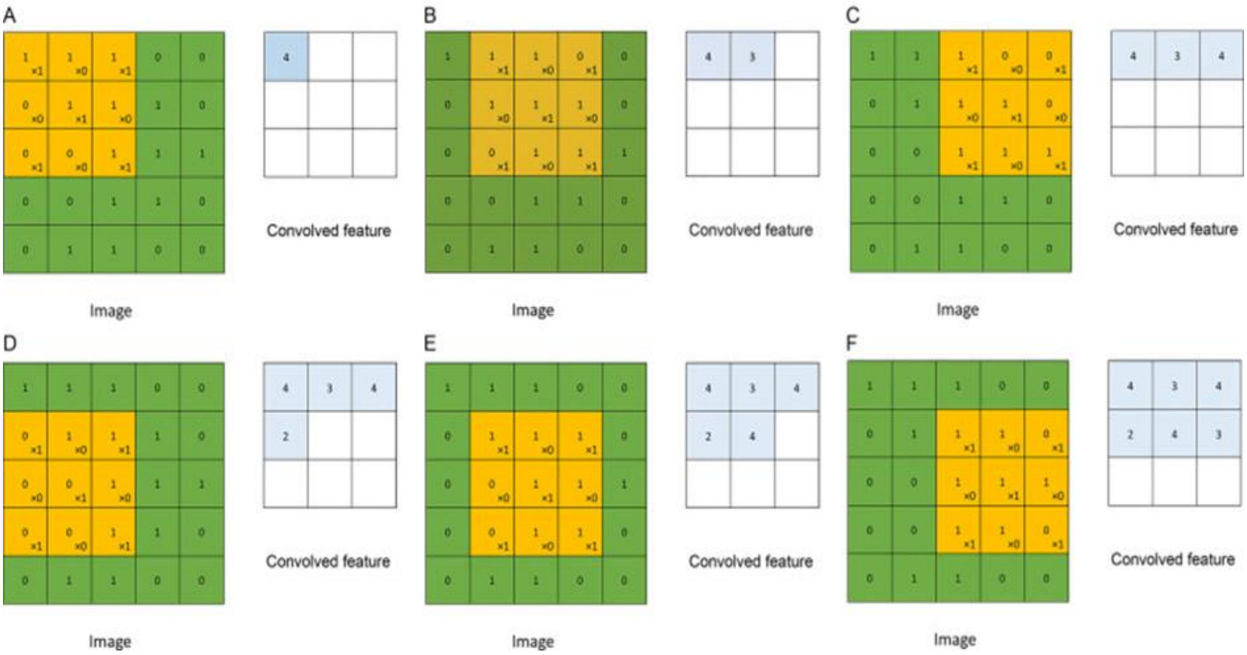

Fig. 3. Functionality of a 2D convolution with a 3x3 pixel core in a 5x5 pixel image.

V. Selection of the Network

The procedure described above was used to set up a model called Visual Geometry Group. Depending on the number of convolutional layers the Model is called VGG16 or VGG19 with sixteen respectively nineteen layers of depth. Even though this approach seemed suited the computing power of the edge device to be used in the tooling machine is limited- for this reason there are better suited computational methods.

According to research done by Microsoft [6] a CNN with higher layer count leads to more training loss. This means that the network could not be further optimized above a certain size. One can imagine that after each convolutional layer, the image is displayed blurrier and blurrier than before until it is so blurry that nothing can be seen. To solve this problem the here used Residual Network (*ResNet*) algorithm was created [7]. *ResNet* builds a sum of the output layer values with the input values after every two convolutional layers. Doing so the input values will be partly retained leading to less training loss in deeper networks.

VI. Pretraining

To save time and enhance the recognition ability a Network with pre-trained parameters will be used [8]. These Networks have already been trained with different Datasets- in this case ImageNet dataset was used by default. The trained parameters are already introduced to the Network to recognize important areas in an image. Since only a small amount of data is available for the contrived problem of recognizing chip windings the network does not need to be trained with huge amounts of sample data to be able to fulfill the planned task of classifying chips. Independent of this the ResNet needs to be trained with specific case dependent data.

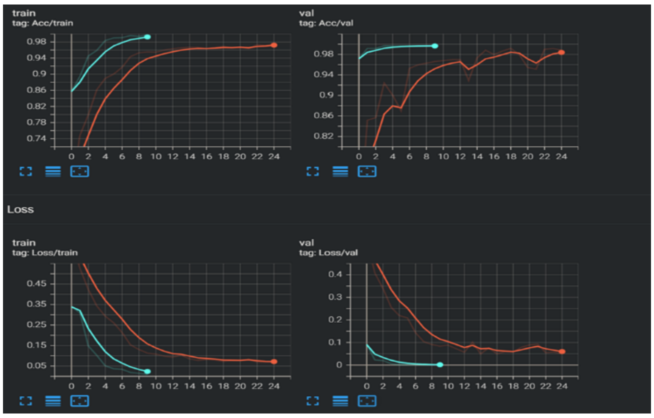

Fig. 4. Graphs for the accuracy (top) and error (bottom) of ResNet during training.

In this process the pre-trained parameters only need to be improved resulting in higher accuracy with shorter training time (see Fig. 4) [9]. It can be seen that the outcome of the pre-trained networks shown in blue is much better than the outcome of the non-pre-trained ResNet shown in red [10]. This outcome is comparable for training data (Fig. 4 left) and validation date (Fig. 4 right). This process is also called transfer learning [11]. A pre-trained network is not only more accurate but also requires less training time.

VII. Data Augmentation

Of great importance even if using a pre-trained network is the amount of training data fed to the network. A large Network is desirable to get good results but with growing Network size also the amount of needed training data grows accordingly [12]. Too little training data for a large network leads to so called overfitting which in consequence leads to worse validation metrics in relation to the training metrics. Since there are only a few images available for the problem of chip tangles this problem can occur very quickly in our specific case. The overfitting is usually expressed by the fact that the network outputs a recognition probability of 100% for some images. This effect is caused by the fact that the model has only learned the training images without really adapting the network to them and delivers wrong results for new images for recognition and classification.

One way to tackle this problem is to artificially generate more images then available which still provides the data needed for an identification of the object to be recognized. This process is well known and is called data augmentation [13]. This involves randomly transforming a portion of the images before training [14]. These transformations can include rotations, distortions, color changes and missing pixels (Fig. 5).

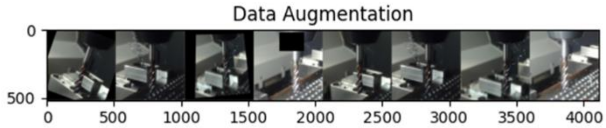

Fig. 5. Example of data augmentation: 4 leftmost images will be augmented leading to the 4 rightmost images (image 1 to 5, 2 to 6, etc.).

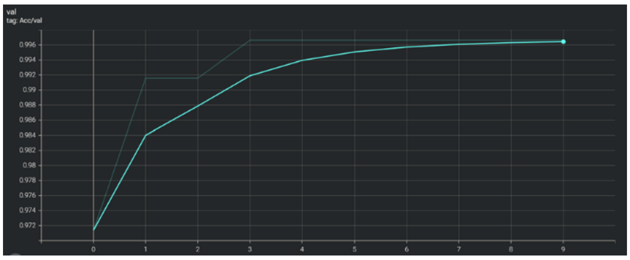

Fig. 6. Classification results depending on training steps.

The outcome of adapting the pre-trained network to some validation images shows that the generated model can provide good classification results (Fig. 6).

VIII. Deployment

In the tooling machine the system was tested in the Machine learning algorithm is executed on an edge device (Craft4EDGE) from Craft4 (https://www.craft4.de/). The Artificial Intelligence has been integrated in the running Linux System on the edge Device parallel to the existing Application Programming Interface making it possible to access the algorithm locally by using the http protocol. A Java-based software from the company Craft4 is used to interrupt the CNC machine in case of chip winding. As a reaction on the unwanted condition the machine changes the direction of the tool rotation- if this is unsuccessful a machine operator will be informed to remove the chip winding manually.

During the tooling process the API of the Edge Device does a cyclic http request to the AI module leading to capturing of an image with the installed camera of the tool which gets evaluated by the AI. In response, the software receives the following information as a JavaScript Object Notation (JSON) stating if unwanted chip buildup was detected or not (Listing 1).

```
{
  "resultClassCode": int | 1 oder 0,
  "resultMessage": string | 1: "Spanwicklung"; 0: "keine Spanwicklung",
  "resultProbability": float | zwischen 0 und 1,
  "speed": float | Zeit in s,
}
```

Listing 1. Code to generate the JSON

IX. Results

The algorithm developed here provides good results after training with presented images and their augmented counterparts to detect unwanted chip buildup which is one of the main causes of tool breakage.

The trained *ResNet* has a training accuracy of 99.78% and a validation accuracy of 99.66% when tested with comparable test images showing that the introduced method is accurate enough for the task of recognizing unwanted chip buildup.

X. Summary

This work shows that it is advisable to use Artificial Intelligence in conjunction with image-processing technology since chip development in a subtractive manufacturing process can be used to draw a variety of conclusions. An evaluation of the image data obtained with a camera in the machine tool by means of an AI-based algorithm is particularly expedient, since such a system can recognize sub-optimal chip production and react accordingly.

Further research must be done to be able to determine more different types of shavings enabling the user to draw more different conclusions using the system.

To be able to detect more pattern the system needs to be trained with more images and different types of scenarios.

The method in its current state is nonetheless very useful to enable the detection of at least one critical type of chips during the tooling process and leads to saved costs and the need of less trained personal to watch the process. Using this system, the production of rejects is reduced leading to significant savings of raw material and time in the process.

With the successful implementation of the project, an important prerequisite for the successful application of image-based manufacturing analytics methods for the detection of excessive tool wear in machine tools has been created.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Roman Radtke conducted and supervised this project. He was responsible for the results and wrote this paper. Joshua Hermann did most of the programming and testing work on site. He also did experiments with different algorithms leading to using *ResNet*. PhD. Aibek Kabanbayev and Prof. Junisbekov Mukhtar Shardarbekovich provided valuable advice and suggestions. Prof. Dr. Alexander Jesser supervised this work from the academic field and put many hints and aspects into this work. All authors had approved the final version.

References

[1] H. Kim, H. Lee, J. S. Kim *et al.*, "Image-based failure detection for material extrusion process using a convolutional neural network," *Int J Adv Manuf Technol*, pp. 1291–1302, 2020.

[2] M. Bernhardt and F. D. S. Ribeiro, and B. Glocker, "Failure detection in medical image classification: A reality check and benchmarking testbed," 2022.

[3] H. Zheng, S. W. A. Sherazi, and S. H. Son, and J. Y. Lee, "A deep convolutional neural network-based multi-class image classification for automatic wafer map failure recognition in semiconductor manufacturing," *Appl. Sci.*, 2021.

[4] L. Liu, W. Ouyang, X. Wang *et al.*, "Deep learning for generic object detection: A survey deep learning for generic object detection: A survey," *Int J Comput Vis*., vol. 128, pp. 261–318, 2020.

[5] M. D. Zeiler and R. Fergus, *Visualizing and Understanding Convolutional Networks*, 2012.

[6] K. Simonyan, *Very Deep Convolutional Networks for Large-Scale Image Recognition*, 2014.

[7] P. S. Huggins, *Deep Residual Learning for Image Recognition*, 2012.

[8] K. He and J. Sun, "Convolutional neural networks at constrained time cost," presented at IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2014, 2015.

[9] C. Tan, *A Survey on Deep Transfer Learning*, 2018.

[10] S. Ren, *Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks*, 2015.

[11] M. Hussain and J. J. B. D. R. Faria, "A study on CNN transfer learning for image classification," presented 18th Annual UK Workshop on Computational Intelligence, Nottingham, June 2018.

[12] M. Shaha and M. Pawar, "Transfer learning for image classification," in *Proc. 2018 Second International Conference on Electronics, Communication and Aerospace Technology*, 2018, pp. 656-660.

[13] A. Mikołajczyk and M. Grochowski, "Data augmentation for improving deep learning in image classification problem," in *Proc. 2018 International Interdisciplinary PhD Workshop (IIPhDW)*, 2018.

[14] L. Engstrom, B. Tran, D. Tsipras, L. Schmidt, and A. Madry, "A rotation and a translation suffice: Fooling CNNs with simple transformations," in *Proc. ICLR Conference*, 2019.

Joshua Hermann was born in Kirchen, Germany.

After A-Levels he worked in the field of predictive maintenance as an Intern in Toronto. Afterwards he started studying mechatronics and robotics at the University of Applied Sciences Heilbronn, Germany. The focus of his studies are machine learning and computer Vision.

Roman Radtke was born in Pinneberg, Germany.

After a training as a precision mechanic he studied medical technology, becoming a Dipl.-Ing. (FH) at the HAW Hamburg, Germany in 2006.

Next, he worked some time in medical research as a project engineer at the AO Research Institute Davos, Switzerland. Following was a position as a support field engineer in Digital X-Ray Imaging and Super-resolution microscopy. As a research assistant at the University of Applied Sciences Heilbronn, Germany, Dipl.-Ing. (FH) Roman Radtke, Ph.D. candidate, he was contributing to several publications in the field of medical technology and industrial internet. He also works as an author for several international magazines since 2013.

Junisbekov Mukhtar Shardarbekovich was born on November 1, 1951 in the city of Dshambul, Kazakhstan. In 1974 he graduated from the V.I. Lenin Kazakh Polytechnic Institute in the field of "Automated control Systems". In 1979 he entered the postgraduate course at the Moscow Textile Institute named after A.N. Kosygin at the Department of Automation and Industrial Electronics, in 1982 he graduated from graduate school in December and was accepted as a teacher at the Dshambul Technological Institute of Light and Food Industry at the Department of Automation and Industrial Electronics. M.S. Junisbekov is the author of 25 textbooks, 4 monographs and more than 150 scientific papers.From 1987-1991 he was the head of the Department of Automation and Electronics, 2001-2009 Head of the Department of Technical Cybernetics. In 2005, M.S. Junisbekov was awarded the academic degree of professor of M.H. Dulati TarSU. Since 2009, he worked as director of the Institute of Postgraduate Education and Advanced Training, 2010-2013. Head of the Department of "Technical Cybernetics". Since 2013, Head of the Department of Automation and Telecommunications.

Kabanbayev Aibek Batyrbekovich is an associate professor at the Dulaty University Taraz, Kazakhstan.

Alexander Jesser was born in Dshambul, Kazakhstan.

He holds the diploma degree in computer engineering from the University of Paderborn, Germany and the Ph.D. in computer engineering from the Johann-Wolfgang Goethe University of Frankfurt a.M., Germany. Since 2013 he is a full professor for embedded systems and communications engineering at the University of Applied Sciences Heilbronn, Germany. In 2019 he became the study dean for the bachelor and the master program Electrical Engineering from the same University. Since 2021 he is the head of the Institute of Intelligent Cyber-Physical Systems (ICPS; www.hs-heilbronn.de/icps) at the University of Applied Sciences in Heilbronn, Germany. He is conducting research in the field of image processing in industrial and medical technology applications. Prior to his academic carrier he worked in leading positions with companies in the medical technology field and in automotive industry.