Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 104 (2021) 1107–1112

www.elsevier.com/locate/procedia

54th CIRP Conference on Manufacturing Systems

# A computer vision system for saw blade condition monitoring

Nicolas Jourdan*,[a], Tobias Biegel[a], Volker Knauthe[b], Max von Buelow[b], Stefan Guthe[b,c], Joachim Metternich[a]

[a]Institute of Production Management, Technology and Machine Tools (PTW), Otto-Berndt-Str. 2, 64287 Darmstadt, Germany
[b]Interactive Graphics Systems Group (GRIS), Fraunhoferstr. 5, 64283 Darmstadt, Germany
[c]Fraunhofer-Institute for Computer Graphics Research (IGD), Fraunhoferstr. 5, 64283 Darmstadt, Germany

* Corresponding author. Tel.: +49 6151 16-24862 ; fax: +49 6151 16-20087. *E-mail address:* n.jourdan@ptw.tu-darmstadt.de

**Abstract**

Tool condition monitoring is a key component of predictive maintenance in smart manufacturing. Predicting excessive tool wear in machining processes becomes increasingly difficult if different materials need to be processed. We propose a novel computer vision-based system for saw blade condition monitoring that is independent of the processed materials and combines deep learning with classic computer vision. Our approach allows for accurate condition monitoring of blade wear which can further be used for predictive maintenance. Additionally, the system can classify different defect types such as missing blade teeth, thus preventing the production of scrap parts.



*Keywords:* Condition monitoring; computer vision; deep learning; saw blade wear;

## 1. Introduction

The state of a cutting tool is a crucial component in any metal cutting process since unplanned downtimes in terms of scrap parts and machine tool breakage are caused by the usage of worn tools [1]. It is estimated that the fraction of downtime that is caused solely by tool breakage amounts to 7-20% [2]. Thus, monitoring the progression of cutting tool wear within the cutting process is required to exchange worn tools in time and consequently avoid unexpected downtime and save costs [3]. Tool Condition Monitoring (TCM) comprises the monitoring of machining tools with the help of sensor processing techniques to reduce the amount of unplanned downtimes due to tool wear or tool damage using hardware for signal acquisition and software for signal analysis and interpretation [4]. The typical sequence of activities in a TCM system comprises sensorial perception and knowledge acquisition, advanced signal processing and decision making which can be considered as the most integral stage and usually involves the application of Artificial Intelligence (AI) methodologies [5]. Traditional approaches to TCM apply so called direct methods (mostly offline) that use vision sensors, proximity sensors, radioactive sensors etc. or indirect (online) methods that measure for instance spindle motor current, cutting force, vibration or acoustic emission and are used to infer the tool condition during machining [6].

Research in the field of TCM, especially with respect to indirect methods, is quite comprehensive and has been actively pursued throughout the last decades resulting in a considerable amount of review papers being published [7, 8]. Although direct methods are capable of capturing actual geometric changes that are caused by tool wear and are more accurate than indirect methods, they are usually very difficult to implement in practice, due to the mostly inaccessible cutting area and the continuous contact between tool and workpiece [3]. Indirect methods on the contrary have the advantage of being less complicated in terms of the required setup and are more suitable for practical applications [9]. However, using indirect methods involves shortcomings that prevent a successful integration into industrial applications. According to [10], the likelihood of sensor failure is almost of the same order of magnitude as machinery failure which in turn requires sensor redundancy to detect such errors. Due to the fact that indirect methods use a plethora of different sensor signals, the system itself must be monitored continuously. Additionally, such methods are developed under laboratory conditions and are thus often not suitable for real world industrial applications due to the assumption of controlled conditions i.e. low susceptibility to noise and uncer-


10.1016/j.procir.2021.11.186

tainty [11]. Other limiting aspects can be considered in the application of traditional AI methods such as shallow neural networks used for decision making support regarding maintenance activities, since they require data for all classes of conditions, provide valid answers only within training range and have to be retrained each time when values move outside the training range due to environmental changes [10]. Limiting factors with respect to the sensors used in indirect methods can be found in [12]. With the recent advances in deep learning, the capabilities in many areas of AI such as image processing and speech recognition experienced a considerable improvement compared to the performance of more traditional AI methods. This direction has been actively researched mainly in the field of indirect methods for TCM [13, 14, 6]. However, using deep learning the application of online direct methods using vision sensors, i.e. cameras to monitor tool wear becomes more interesting, especially when there is a process that does not rely on the existence of a cooling lubricant that would prevent a camera to clearly identify the tool while processing [9]. In recent years researchers started to adress the application of computer vision to monitor tool wear more intense, see [12] for a review of different methods. Dutta et al. [15] proposed a method to predict progressive tool wear using extracted features from turned surface images via gray level co-occurrence matrix (GLCM), Voronoi Tesselation (VT) and discrete wavelet transform (DWT). Ong et al. [16] used a wavelet neural network (WNN) for tool wear monitoring in CNC end milling process of high speed steel. Sun and Yeh [17] developed an on-machine turning tool insert condition monitoring system that can identify four external tool insert conditions, namely fracture, built-up edge, chipping and flank wear. However, none of the related work uses deep learning to improve the performance of the computer vision based systems.

The presented paper can be considered as a follow up to the indirect method proposed by Busse et al. [18], in which the authors used a nonlinear autoregressive with exogenous inputs (NARX) neural network [19] to predict bandsaw failure based on sensor values. However, the system developed by Busse et al. is dependent on the material that is being processed on the band saw, since the sensor values strongly differ given the characteristics of the material. In modern industrial applications, bandsawing is the primary process for the preparation of raw materials [20] and thus processing various kinds of material would require specific failure curves for each of the materials being processed, which results in an unfeasable requirement for practical purposes. The main objective of the proposed approach is to combine traditional methods in computer vision with recent advances in deep learning to allow for more sophisticated saw blade condition monitoring that is independent of the material being processed and thus requires less data and training efforts leading to increased flexibility for industrial applications. In order to achieve this objective, we follow a twofold approach to design our system: In the first step, we employ the deep neural network architecture YOLOv3 [21], which is an extension of the YOLO network presented in [22], to be able to detect blade teeth. The second step embodies the detection of the backside of the saw blade, which is accomplished by using a Sobel filter and the random sample consensus (RANSAC) algorithm [23]. Given the respective blade teeth and the backside of the saw band we can determine the height of the saw blade itself by calculating the perpendicular distance between these instances. The approach can be run online during active machining using a short exposure time on the camera to limit motion blur. Figure 1 illustrates the height measurements on a test image retrieved by applying the aforementioned twofold approach.

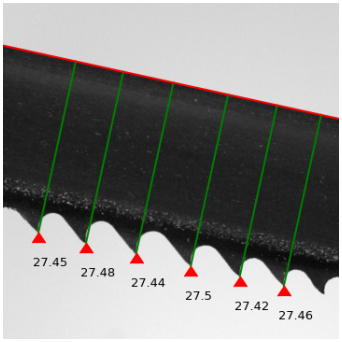
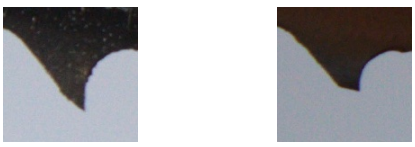



Fig. 1: Examplary result image of the detections on an image from the test-set used in the experiments. The red triangle markers show the detections of the saw blade teeth. The red line highlights the saw blade back line.

The remainder of the paper is organized as follows: In Section 2 we present the methodology of our computer vision-based system for saw blade condition monitoring. Consequently, Section 3 includes the experiments and validation of our approach. The conclusion and discussion of future research form Section 4 of our paper.

## 2. Methodology

In our study, we focus on the popular Kasto SBA2 model as our test-bed. An example of the typical flank wear observed on the blade of this saw is depicted in Figure 2. As the average total height difference between an unused blade and a worn saw blade that needs to be exchanged only amounts to about 0.59 mm in our experiments, the optical measurements need to be very precise to produce usable results. The overall methodology is depicted in Figure 3. The computation pipeline can be divided into the detection of the saw blade back and the saw blade teeth. Both can be computed in parallel to save computation time. The proposed system is implemented in Python. YOLOv3 is implemented using the deep learning library Keras [24], while the image pre-processing and Sobel filter operations are implemented using OpenCV [25].



(a) Unused saw blade tooth (b) Worn saw blade tooth

Fig. 2: Examples for the visual appearance of saw blade teeth on an unused and a worn blade. The height difference between (a) and (b) is 0.59 mm.

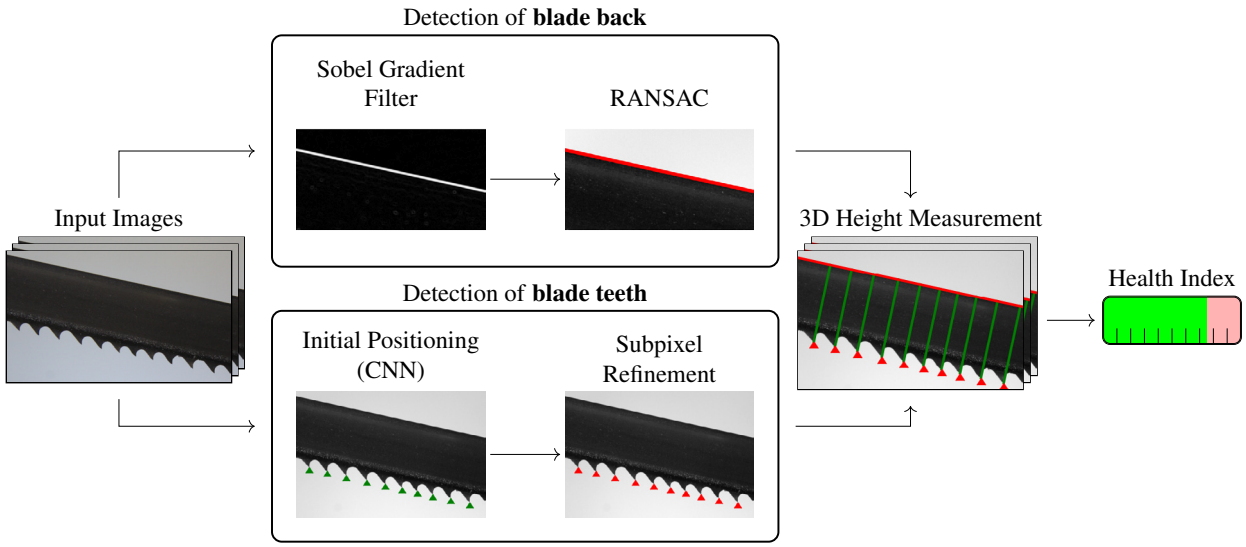
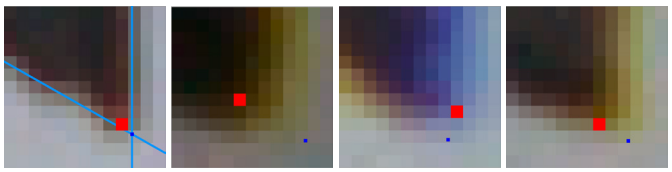



Fig. 3: Overview of the pipeline used for estimating the Health Index of the saw blade.

### 2.1. Detection of blade teeth

The saw blade teeth positions within the image are determined in two stages. First, we utilize an object detection Convolutional Neural Network (CNN) for detecting the saw blade teeth and providing an initial estimate for their positions. Second, we apply sub-pixel refinement to increase the accuracy of the position estimates.

*Initial positioning.* We utilize the popular YOLOv3 CNN architecture [21, 22] for the initial position estimation of the saw blade teeth. As YOLOv3 was originally designed for object detection with bounding boxes, we slightly simplify the architecture for the keypoint detection usecase on hand. As we do not require the detector to be scale invariant, we only use the last feature layer for prediction. Further, we do not require information about the spatial extent of the saw blade teeth. Thus, we fix the size of the anchor boxes as well as the predicted bounding boxes to a size of $11 \times 11$ pixels with the blade tips in the very center. To prevent overfitting as well as to improve overall performance, data augmentation is used. *Random Cropping*, *Left-Right Flip*, *Brightness* as well as *Salt and Pepper Noise* are applied with a probability of $p = 0.5$ during training. The data augmentation is applied online during training. Hence, it does not affect the number of available training images, but rather randomly alters the existing images in every epoch.



(a) Close estimation (b) Far off estimation (c) Left correction (d) Right correction

Fig. 4: Four examples for sub-optimal annotations (red) in comparison to sub pixel accuracy (blue). On average the distance between the red center location and blue locations is 1.55 *px* in our dataset. The best case is close to zero, limited by the nature of subpixels and manual labeling. The worst case is 8.25 *px*. The lines in (a) show two exemplary tangents which are used to calculate the sub-pixel position.

*Sub-pixel refinement.* Our algorithm's blade tooth position approximation is trained on manually generated annotations. These manual annotations, as illustrated in Figure 4, may be subject to bias due to individual and inconsistent annotations on greyscale gradients that represent an edge [26]. This is especially the case for multiple annotators. Additionally, it is not feasible to train a network with every possible ground truth, which leads to non-perfect predictions in unknown data. In order to make results explainable and more robust, the algorithm executes an additional refinement step for every single tooth. Therefore, we use the sub-pixel approximation of Förstner and Gülch [27], which uses a search window $W$ centered around an initial guess to refine a point $p$ sensitive to edges in the texture of the given image $I$. The whole optimization is shown in Equation (1).

$$\hat{p} = \arg\min_{p} \int_{q \in W} \left( \nabla I_q^T (p - q) \right)^2 dq \tag{1}$$

This approach finds the desired optimal two-dimensional corner point $\hat{p}$. To achieve this it optimizes $p$ such that each tangent line $\nabla I_q^T (p - q) = 0$ at $q$ within a search window $W$ of 5 pixel radius mutually intersects and is orthogonal to the image gradient at $q$. The resulting point $\hat{p}$ is the point with the minimal distance to all intersections of the weighted tangents.

A simplified example with two tangents can be seen in Figure 4a. The tangent line is defined by a line that goes through $p$ and $q \in W$. The transposed image gradient $\nabla I_q^T$ multiplied by the vector $p - q$ effectively forms a dot product, which weights the tangent lines laying *on* the image gradient (i.e. being orthogonal to the gradient direction). Due to this weighting, tan-

gent lines that correspond to the image texture resulting in the corner point are represented stronger in the optimization. This combines the strengths and weaknesses from both approaches most importantly the very fast global initial saw tooth localization and the precise reproduceable local sub-pixel refinement. Additionally, this procedure eliminates human labeling bias and corrects slightly off predictions from the network.

### 2.2. Detection of blade back

The position and angle of the saw blade back edge are determined using classic computer vision methods. First, an image crop is extracted that focusses on the transition between the blade and the image background as the primary contrast in the image. A gaussian blur is consequently applied to the crop to reduce noise and unnecessary detail on the blade material. Next, a Sobel operator [28] with a kernel size of $5 \times 5$ is utilized to measure horizontal and vertical gradients in the image. Using the resulting gradients $\mathbf{G}_x$ and $\mathbf{G}_y$, the corresponding angles are computer as $\mathbf{\Theta} = \arctan 2\left(\mathbf{G}_x, \mathbf{G}_y\right)$. The gradient angle and magnitude are thresholded to identify candidate pixels that are likely positioned on the the edge of the saw blade back. Finally, the RANSAC algorithm [23] is used for a robust linear regression through the identified edge pixels.

### 2.3. 3D height measurement

*Transformation to real-world coordinates.* After detecting the saw blade teeth positions as well as the saw blade back edge in the image, the pixel-level coordinates $u_i, v_i$ are transformed to real-world 3D coordinates $x_i, y_i, z_i$ to be able to measure the remaining height of the saw blade teeth in milimeters. The projection of the real-world coordinates on the undistorted image plane is described using a pinhole camera model where $f_x, f_y$ and $c_x, c_y$ describe the focal length and principal point coordinates respectively. As the plane in which the saw blade moves is mounted parallel to the camera sensor in a fixed distance $z_{\text{cam}}$, one can obtain the real-world coordinates as

$$x_i = \frac{(u_i - c_x)\, z_{\text{cam}}}{f_x}, y_i = \frac{\left(v_i - c_y\right) z_{\text{cam}}}{f_y}, z_i = z_{\text{cam}}. \tag{2}$$

*Height measurement.* We define the relevant height of a saw blade teeth as the shortest distance between a saw blade teeth described by its coordinates $x_i, y_i, z_{\text{cam}}$ and the saw blade back line described by two points $P_1, P_2$ that lie on it. Thus, the distance is given by a line that is perpendicular to the saw blade back and intersects the saw blade teeth coordinates. The estimated height $\hat{h}_i$ of saw blade tooth $i$ is therefore defined as

$$\hat{h}_i\left(P_1, P_2, (x_i, y_i)\right) = \frac{\left|(y_2 - y_1)\, x_i - (x_2 - x_1)\, y_i + x_2 y_1 - y_2 x_1\right|}{\sqrt{(y_2 - y_1)^2 + (x_2 - x_1)^2}}. \tag{3}$$

As the 3D transformation depends on the accuracy of the intrinsic camera calibration as well as the manually measured distance between the camera and the saw blade $z_{\text{cam}}$, there may be measurement errors that influence the blade height estimation $\hat{h}_i$. Using the assumption that the influence of these measurement errors on the estimation is linear, we fit a linear regression $h_i = \beta_0 + \beta_1 * \hat{h}_i$ on the estimated values using the *training set* to correct for scale and offset errors. As the saw blade wear will not affect all teeth uniformly, we measure multiple saw blade teeth to accurately estimate the current condition of the blade. The individual measurements of $m$ teeth are aggregated by calculating the mean height $\bar{h} = \frac{1}{m} \sum_{i=0}^{m} h_i$.

### 2.4. Health index estimation and further model capabilities

The final step of the computational pipeline is to compute the Health Index (HI) of the saw blade in per-cent. This is done by linear interpolation of the current mean saw blade height $\bar{h}_t$ between the mean height of an unused blade $\bar{h}_{\text{new}}$ (100%) and the minimum usable height $\bar{h}_{\text{def}}$ (0%), $\text{HI}_t = 100\% \cdot \frac{\bar{h}_t - \bar{h}_{\text{def}}}{\bar{h}_{\text{new}} - \bar{h}_{\text{def}}}$. While the estimation of the blade flank wear is the primary objective of this study, it is possible to use the proposed model to detect other abnormalities such as broken teeth which lead to quality degradation and even defects on the machine itself. As the total difference in height between unused and fully worn blades only amounts to 0.59 mm, broken teeth can easily be detected by using a threshold $h_{\text{thresh}} << \bar{h}_{\text{def}}$ on the estimated blade height.

## 3. Experiments

The experiments for this study were conducted using the industrial bandsaw depicted in Figure 5. To validate the proposed methodology, five different levels of wear are physically measured and consequently compared to the wear levels estimated by the condition monitoring system. Thus, a given saw blade was measured in its unused state to define $\bar{h}_{\text{new}}$, its defect state $\bar{h}_{\text{def}}$ as well as three states in between the two extremes. The minimum usable height of a blade $\bar{h}_{\text{def}}$ was determined by experiments using 50 mm square aluminum bars as raw material.

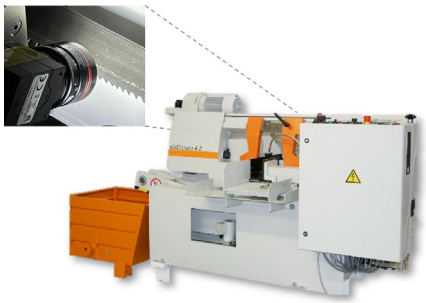


Fig. 5: Kasto SBA2 bandsaw used for the experiments. The industrial camera FLIR BFS-U3-27S5M-C with dimensions 29x29x30 mm is mounted on the back side of the saw blade, which allows to capture and process images during machining. The exposure time is set to $20\mu s$ to limit motion blur.

| State | True blade condition | | Predicted blade condition | | | | Error Confidence Interval (95%) | |
|---|---|---|---|---|---|---|---|---|
| | $HI_t^{GT}$ (%) | $h_t^{GT}$ (*mm*) | $\bar{HI}_t$ (%) | $\bar{h}_t$ (*mm*) | $\sigma_{\bar{HI}_t}$ (%) | $\sigma_{\bar{h}_t}$ (*mm*) | $\bar{HI}_t$ (%) | $\bar{h}_t$ (*mm*) |
| 0 - New blade | 100.00 | 27.33 | 99.56 | 27.33 | 0.49 | 0.0029 | [−0.70, 1.41] | [−0.0041, 0.0083] |
| 1 | 62.71 | 27.11 | 59.38 | 27.09 | 0.95 | 0.0056 | [−4.53, −0.56] | [−0.0267, −0.0033] |
| 2 | 42.38 | 26.99 | 46.45 | 27.02 | 0.66 | 0.0039 | [3.43, 6.29] | [0.0202, 0.0371] |
| 3 | 18.64 | 26.85 | 20.42 | 26.87 | 1.51 | 0.0089 | [0.67, 5.80] | [−0.0041, 0.0342] |
| 4 - Worn blade | 0.00 | 26.74 | 0.00 | 26.74 | 2.19 | 0.0129 | [−3.91, 5.49] | [−0.0231, 0.0324] |

Table 1: Experiment results on the hold-out *test set* for five increasing levels of wear. The condition estimates are computed using a sample size of $n = 15$ images each. Scale and offset parameters are calibrated on a separate validation set.

As the definition of the minimum usable state is qualitative and difficult to define absolute, it is a parameter that must be chosen according to the quality requirements of the respective process step and product. To build a dataset for training and evaluation of the proposed method, 50 images are captured of every state, resulting in a dataset of 250 images. The images are captured from a fixed distance $z_{\text{cam}}$ parallel to the saw blade. Further, the blade height was measured at 10 different positions across the band and consequently averaged for every state to define accurate ground truth labels for the model. The generated dataset is split into a *training set* using 30 images per state, a *validation set* using 5 images per state as well as a *test set* using 15 images per state. Consequently, the datasets are manually labeled with keypoint annotations at the blade tooth tips for training and evaluation of the YOLOv3 detections. The YOLOv3 model is trained for a total of 80 epochs using the *training set* with batch size 8 and at an initial learning rate of 0.006 which is decreased to 0.0005 after 50 epochs. The hyperparameters, including the bounding box size, batch size, learning rate schedule, and epoch count, are tuned using the *validation set*.

The results of the initial blade teeth detection using YOLOv3 are listed in Table 2. The trained model is able to accurately detect the blade teeth, with a Mean Absolute Error (MAE) of 1.3466 $px$ on the *test set* for all detections that are matched to the ground truth with an euclidian error $\leq 3px$ (93%). To analyze the detection performance, we further report the Average Precision (AP), using an euclidian error $\leq 3px$ as a matching criterion for true positives. We employ the euclidian error as a matching criterion as this better represents the usecase compared to intersection over union matching. Further, it is observable that the performance difference between *test set* and *training set* is rather small, indicating that there is no signficant overfitting problem present. As the variations in the blade and consequently the images thereof are very small, overfitting is not a significant problem in this setting, especially when compared to other application areas with much higher intra-class variations such as autonomous driving. The quantitative experiment results of the complete condition monitoring pipeline are listed in Table 1. The experiments have been conducted using a AMD Ryzen 9 3900X processor. The processing time per image amounts to ~397ms using a GeForce RTX 2080Ti GPU and ~6,12s using only the CPU. Taking into account that the saw blade wears slowly, the processing time of our algorithm is thus considered sufficiently fast for being used in an industrial application. Notably, the model is able to estimate the blade flank wear at all five wear states within sub-millimeter accuracy. To provide a practical evaluation of the performance as well as statistical guarantees, we further analyze the distribution of the measurement errors of the model on the *test set* in Table 1. To test the $H_0$ hypothesis that the regression errors follow a normal distribution, a normality test based on D'Agostino and Pearson [29] is conducted. The resulting p-value $p = 0.42$ well exceeds the significance level $\alpha = 0.05$, thus $H_0$ cannot be rejected. Interestingly, the analysis shows a trend of increasing standard error of the mean of the Health Index estimate $\sigma_{\bar{HI}_t}$ for increasing levels of wear. This result is to be expected as not all flanks will wear uniformly while little variance is expected for an unused blade. Finally, we provide the confidence intervals for the regression error concerning the Health Index $\bar{HI}$ as well as the blade height $\bar{h}_t$ for each state which is estimated using a sample size of $n = 15$ respectively. Similarly to $\sigma_{\bar{HI}_t}$, the width of the confidence intervals increases with increasing wear as well. While this accuracy is already usable for condition estimation, the confidence interval can be further tightened by increasing the sample size $n$ used for estimation if necessary for the respective usecase. In order to facilitate the interaction between the system and the machine operators, a web interface has been developed, as shown in Figure 6 on the next page, that displays the HI of the saw blade on a screen, during machining.

| Split | $MAE_{d\leq 3px}$ ($px$) | $AP_{\leq 3px}$ (%) |
|---|---|---|
| Training Set | 1.2609 | 88.7 |
| Validation Set | 1.2894 | 88.1 |
| Test Set | 1.3466 | 88.5 |

Table 2: YOLOv3 detection results on the dataset splits, averaged over all levels of wear.

## 4. Conclusion

In this paper, we proposed a novel online direct TCM method for robust condition monitoring of saw blade flank wear using computer vision and deep learning. The processing pipeline of the proposed method estimates the position and angle of the blade back via classic computer vision. The blade teeth in turn are initially detected by deep learning and further refined to sub-pixel accuracy. We combine both to a 3D measurement of the blade height which is used to compute the Health Index of the blade. Our experiments show, that the

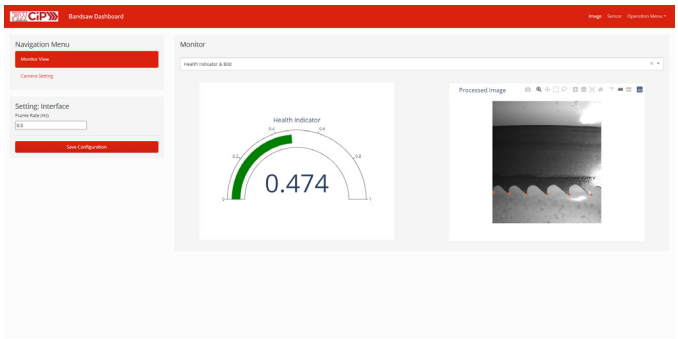



Fig. 6: Web interface based on which the machine operator can interact with the system. The health indicator as well as the image and corresponding detections are visualized. The degradation can also be plotted over time in a separate view.

proposed method is able to accurately estimate the blade wear while being robust against changes in material characteristics and sensor drift by design. The proposed method further requires only little training data, which renders the application in a productive environment feasible with low initial investment. While our approach already provides usable condition estimates, there are several possibilities for future work. Most importantly, flank height decrease might not be the only symptom of wear on a saw blade. Further experiments are necessary to determine if there are other important types of wear that are of interest for a condition monitoring system. Lastly, the classification and regression capabilities of the CNN detection framework can be exploited further, possibly allowing for direct Health Index prediction.

*Acknowledgements.* Part of the research in this paper was funded by DFG (Deutsche Forschungsgemeinschaft) project 407714161.

## References

[1] Atli, A.V., Urhan, O., Ertürk, S., Sönmez, M., 2006. A computer vision-based fast approach to drilling tool condition monitoring. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture 220, 1409–1415.

[2] Dan, L., Mathew, J., 1990. Tool wear and failure monitoring techniques for turning—a review. International Journal of Machine Tools and Manufacture 30, 579–598.

[3] Ambhore, N., Kamble, D., Chinchanikar, S., Wayal, V., 2015. Tool condition monitoring system: A review. Materials Today: Proceedings 2, 3419–3428.

[4] Zhang, C., Yao, X., Zhang, J., Jin, H., 2016. Tool condition monitoring and remaining useful life prognostic based on a wireless sensor in dry milling operations. Sensors 16, 795.

[5] Teti, R., Jemielniak, K., O'Donnell, G., Dornfeld, D., 2010. Advanced monitoring of machining operations. CIRP annals 59, 717–739.

[6] Serin, G., Sener, B., Ozbayoglu, A., Unver, H., 2020. Review of tool condition monitoring in machining and opportunities for deep learning. The International Journal of Advanced Manufacturing Technology , 1–22.

[7] Ahmad, M.I., Yusof, Y., Daud, M.E., Latiff, K., Kadir, A.Z.A., Saif, Y., 2020. Machine monitoring system: a decade in review. The International Journal of Advanced Manufacturing Technology , 1–15.

[8] Mohanraj, T., Shankar, S., Rajasekar, R., Sakthivel, N., Pramanik, A., 2020. Tool condition monitoring techniques in milling process—a review. Journal of Materials Research and Technology 9, 1032–1042.

[9] Zhu, K., San Wong, Y., Hong, G.S., 2009. Wavelet analysis of sensor signals for tool condition monitoring: A review and some new results. International Journal of Machine Tools and Manufacture 49, 537–553.

[10] Grimmelius, H.T., Meiler, P.P., Maas, H.L., Bonnier, B., Grevink, J.S., van Kuilenburg, R.F., 1999. Three state-of-the-art methods for condition monitoring. IEEE Transactions on Industrial Electronics 46, 407–416.

[11] Shi, C., Panoutsos, G., Luo, B., Liu, H., Li, B., Lin, X., 2018. Using multiple-feature-spaces-based deep learning for tool condition monitoring in ultraprecision manufacturing. IEEE Transactions on Industrial Electronics 66, 3794–3803.

[12] Dutta, S., Pal, S., Mukhopadhyay, S., Sen, R., 2013. Application of digital image processing in tool condition monitoring: A review. CIRP Journal of Manufacturing Science and Technology 6, 212–232.

[13] Goodfellow, I., Bengio, Y., Courville, A., Bengio, Y., 2016. Deep learning. volume 1. MIT press Cambridge.

[14] Zhao, R., Yan, R., Chen, Z., Mao, K., Wang, P., Gao, R.X., 2019. Deep learning and its applications to machine health monitoring. Mechanical Systems and Signal Processing 115, 213–237.

[15] Dutta, S., Pal, S.K., Sen, R., 2016. Tool condition monitoring in turning by applying machine vision. Journal of Manufacturing Science and Engineering 138.

[16] Ong, P., Lee, W.K., Lau, R.J.H., 2019. Tool condition monitoring in cnc end milling using wavelet neural network based on machine vision. The International Journal of Advanced Manufacturing Technology 104, 1369–1379.

[17] Sun, W.H., Yeh, S.S., 2018. Using the machine vision method to develop an on-machine insert condition monitoring system for computer numerical control turning machine tools. Materials 11, 1977.

[18] Busse, A., Meudt, T., Metternich, J., 2017. Einsatz digitaler Systeme zur Prozessüberwachung. ZWF Zeitschrift für wirtschaftlichen Fabrikbetrieb 112, 652–657.

[19] Kumpati, S.N., Kannan, P., et al., 1990. Identification and control of dynamical systems using neural networks. IEEE Transactions on neural networks 1, 4–27.

[20] Saglam, H., 2011. Tool wear monitoring in bandsawing using neural networks and taguchi's design of experiments. The International Journal of Advanced Manufacturing Technology 55, 969–982.

[21] Redmon, J., Farhadi, A., 2018. Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767 .

[22] Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2016. You only look once: Unified, real-time object detection, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 779–788.

[23] Fischler, M.A., Bolles, R.C., 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM 24, 381–395.

[24] Chollet, F., et al., 2015. Keras. https://keras.io.

[25] Itseez, 2015. Open source computer vision library. https://github.com/itseez/opencv.

[26] Lutnick, B., Ginley, B., Govind, D., McGarry, S.D., LaViolette, P.S., Yacoub, R., Jain, S., Tomaszewski, J.E., Jen, K.Y., Sarder, P., 2019. An integrated iterative annotation technique for easing neural network training in medical image analysis. Nature machine intelligence 1, 112–119.

[27] Förstner, W., Gülch, E., 1987. A fast operator for detection and precise location of distinct points, corners and centres of circular features, in: Proc. ISPRS intercommission conference on fast processing of photogrammetric data, Interlaken. pp. 281–305.

[28] Sobel, I., Feldman, G., 1968. A 3x3 isotropic gradient operator for image processing. a talk at the Stanford Artificial Project in , 271–272.

[29] Pearson, E.S., D "'AGOSTINO, R.B., Bowman, K.O., 1977. Tests for departure from normality: Comparison of powers. Biometrika 64, 231–246.