



Automated geometric analysis of metallic components through picture recognition models for manufacturing technology assessments

Tobias Buechler¹ · Martin Schulz¹ · Michael F. Zaeh¹

Received: 31 March 2022 / Accepted: 4 July 2022 / Published online: 26 July 2022
© The Author(s) 2022

Abstract

The selection and interaction of various manufacturing technologies are key difficulties in product development and production processes. A component's geometry is one of the most important factors to consider when choosing the best technology. This article presents a method for an automated geometry analysis of metallic components. The goal is to analyze manufacturing technology alternatives regarding their capability to create required geometries. It also aims at short computing times since the outcome of this geometric analysis supplements a part screening methodology for the selection of the most suitable manufacturing technology for each component. To achieve a successful classification, artificial intelligence (AI) approaches are trained with images of the components that are labeled with suitable manufacturing technologies. The AI models hence learn how components of different manufacturing technologies look like and which characteristics they embody. To support the classification model, object recognition models are tested to automatically extract component features such as holes, coinages, or profile compositions. After training and comparing different AI approaches, the best performers are selected and implemented to analyze unseen image data of upcoming projects. In summary, this article's research unifies existing AI approaches for image analyses with the field of production technology and product development. It provides a general methodology for applying image classification and object detection approaches in development processes of metallic components.

Keywords AI-based picture recognition and classification · Object detection · Part screening · Manufacturing system design · Early-stage flexibility

1 Introduction

Components' manufacturing technologies must be determined early in product development since they significantly impact the subsequent design and production steps [1]. So far, this decision is often made based on previous product generations and the stakeholders' experiences, potentially causing a human bias towards the status quo. An objective manufacturing technology comparison is thus essential to accelerate product development processes and to minimize potential change costs for correcting wrong decisions at a later point in time [2]. Furthermore, the technology assessment must be applicable early in product development whilst

ensuring a high degree of automatization to handle large quantities of components to analyze. This article and the method for geometric analyses support the trend of achieving shorter development times in increasingly competitive industries, e.g., the automotive sector [3]. In the frame of this article and its application example, five manufacturing technologies are relevant and compared: deep drawing, casting, pressing, rolling, and additive manufacturing. A holistic comparison of potential manufacturing technologies on a component level needs to consider the following aspects [4, 5]:

- the requirements that the components must fulfill regarding e.g., corrosion or temperature
- the impact of components' manufacturing technologies on subsequent production steps of the final product e.g., joining complexity
- the costs over the components' lifecycles depending on the chosen manufacturing technologies

✉ Tobias Buechler
tobias.buechler@iwb.tum.de

¹ Institute for Machine Tools and Industrial Management, Technical University of Munich, 80333 Munich, Germany

- the required geometric characteristics of each component

Each of these aspects is addressed by one of four so-called *modules* that provide an overarching *part screening methodology for a manufacturing technology selection* with relevant data for comparing the technology scenarios (see Fig. 1). The modules derive and predict data to fill information gaps in early product development stages and thus enable an evaluation of technology alternatives for each component. The screening methodology evaluates these alternatives based on fuzzy rule sets that represent the strengths and weaknesses of each manufacturing technology. Three modules have already been completed and published [4, 6]: The *requirements module* derives requirements based on the components' positions in the final product. The *production module* analyzes the logistics and production inferences for different manufacturing technologies. Based on that, the *cost module* predicts and calculates manufacturing and logistics costs as well as necessary investments in the production line of the final product for each manufacturing scenario.

As mentioned in the fourth bullet point above, assessing the components' geometries is an essential pillar when comparing manufacturing technologies. Building this pillar is thus the aim of the work presented in this paper. It elaborates on the *Geometry Module*, the fourth source of information for the part screening methodology. The geometry module aims at an automated geometry analysis of metallic components to support the manufacturing technology assessment: first, an image database is built and subsequently labeled with information regarding the components' suitable manufacturing technologies. Based on that, different AI approaches are trained and compared to classify component

images by manufacturing technology. The resulting percentage values per technology class represent geometric similarities with the characteristics of different manufacturing technologies. These similarity values thus indicate the suitability of technologies to manufacture the required geometry. This picture-based approach aims at short computing times within the overarching part screening methodology. Furthermore, an object recognition model identifies and counts component features such as holes or coinages within the pictures. These features support the classification approach by supplying additional information.

The article first deduces the research gap in Sect. 2 and then presents the general approach for automated geometric analyses of metallic components in Sect. 3. Section 4 compares the performance of the different AI models within an application example.

2 Geometric analyses on a component level

This section first introduces fundamentals (Sect. 2.1) for the understanding of the article, then focuses on the analysis of existing approaches for picture recognition and geometric component evaluation in Sect. 2.2. Based on that, the research gap is deduced in 2.3 to motivate this article's research.

2.1 Fundamentals

This section introduces the fundamental theory for understanding the different classification and object detection approaches of this paper.

Convolutional neural networks. Convolutional neural networks (CNN) are relevant for the understanding of the geometry module since they have established themselves as successful model architectures in picture classification [7, 8]. A layer of a CNN usually consists of three processing steps: The first step is the extraction of features out of pictures using *filters* or *kernels* [6]. After that, non-linear activation functions such as the *ReLU* function (rectified linear unit) are used to convert the linear output of the convolution to non-linear values. In a third step, the initial dimensions are adjusted by using so-called *pooling functions*. A pooling function replaces the output of the network with a summary of surrounding neurons. *Max pooling* determines only the maximum value from a selection of neurons. Another function is *average pooling* that calculates the mean value of the neurons and passes it on to subsequent layers [7]. This reduction in dimensionality and complexity minimizes the required computing power. The different pooling methods are illustrated in Fig. 2.

The structure of a simple CNN is depicted in Fig. 3. The network begins with a layer of 28×28 input neurons that are

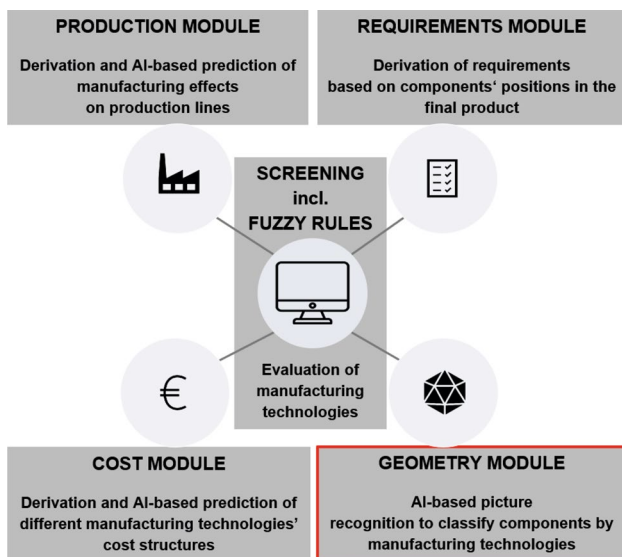


Fig. 1 The four modules of the screening methodology, according to Buechler et al. [4]

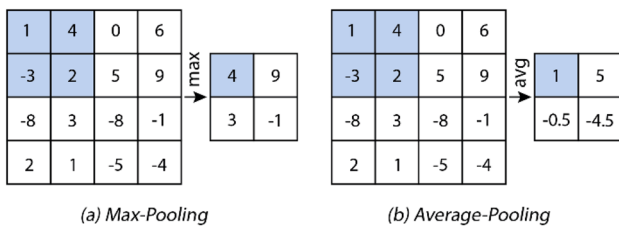


Fig. 2 Exemplary feature maps and relevant pooling methods according to Paass et al. [8]

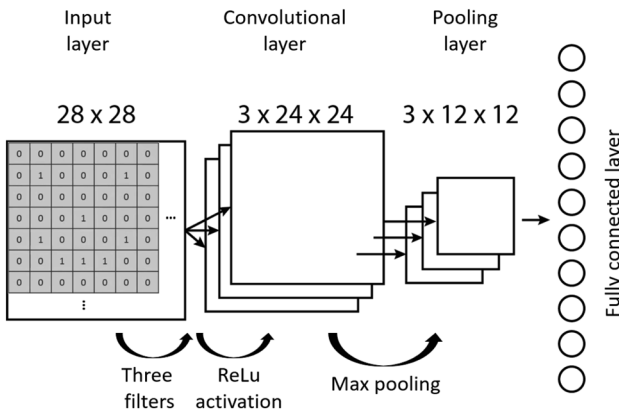


Fig. 3 Exemplary architecture of a CNN with three processing steps and ten classes, according to Priftis et al. [10]

used to encode the pixels of an image, e.g., as binary values for dark or bright areas. Three filters are overlappingly applied to the input layer and processed through a ReLU activation. This convolution results in three numeric feature maps with $3 \times 24 \times 24$ neurons. Feature maps allow for insights, e.g., which features the CNN detects and indicates. Subsequently, max pooling with a dimension of 2×2 pixels across each of the three feature maps creates another hidden layer with $3 \times 12 \times 12$ neurons. The last layer of the network is a *fully connected layer*, each neuron from the max-pooling layer is thus connected to each of the exemplary output neurons that represent specific classes, e.g., manufacturing technologies. When training a CNN, the weights of each layer as well as the number and numeric structure of the filters are adapted to achieve the best classification results [9, 10]. So-called *full-stack AI-platforms* offer pre-trained architectures to support the implementation of CNNs through the benefits of *transfer learning* [11].

Fuzzy logic. The outcome of the Geometry Module supplements the fuzzy rules of the part screening methodology. Fuzzy logic imitates the human behavior to perceive information in fuzzy ranges like *low*, *medium*, and *high*, e.g. temperature [12]. Input parameters are first *fuzzified* via membership functions, e.g., as a triangular or a trapezoidal set-up. The numeric output of a fuzzy set is generated by the

defuzzification using so-called fuzzy rules [13]. More details and examples can be found in Sect. 4.3.

2.2 State of the art

The following paragraphs elaborate on existing classification and object detection methods to ensure an understanding of the geometry module’s approach. There are approaches that derive 3D descriptors from synthetic CAD-models for a classification of unseen objects [14]. However, this article focuses on testing an image-based classification to achieve short computing times.

Picture recognition and classification

General approaches. Picture recognition is widely used, for emotion detection, quality control in production systems or even weather forecasting [15, 16]. Two architectures are currently of high relevance for picture recognition and will therefore be briefly introduced: The *VGG* (visual geometry group) architecture was developed as part of the *ImageNet* competition in 2014 and achieved high classification accuracies [17]. The *ResNet* architecture was the first picture classification approach to beat humans in terms of classification accuracy [8]. In both approaches, CNNs are used to classify image data. Besides high accuracies, a current challenge is the distinction of objects from the background. Furthermore, the definition of the classes must ensure a sufficient demarcation from each other, e.g., regarding the shape and color of the objects [8]. In the first layers, simpler image features such as edges, lines, or corners are recognized in convolutional layers. Within additional layers, the characteristics become more complex and often exceed human perception. The features that are recognized in the filters depend on the field of application and differ for each model [8].

Analyzing 3D objects with pictures. The analysis of three-dimensional objects using two-dimensional images offers some advantages over direct 3D CAD file analysis, e.g., shorter computing times. However, the biggest challenge is to draw conclusions based on two-dimensional images for the three-dimensional world [18, 19]. To overcome this challenge, multiple views or perspectives can be used to grasp the complexity of the examined objects [20]. Using multiple views to analyze a single object imitates the natural human behavior of looking at objects from several perspectives [20]. Widespread architectures such as *ResNet* [21] or *VGG* can be used for 2D-image analyses whilst ensuring good performances [22].

Figure 4 illustrates another methodology developed by Su et al. [19]: the generated 2D views of a 3D model are first individually analyzed using one single CNN (CNN1). In the next step, the discovered features are bundled across all views within a so-called *View Pooling* layer to create a *multi-view* approach. Based on that, the classification is carried out by a second CNN (CNN2). This methodology

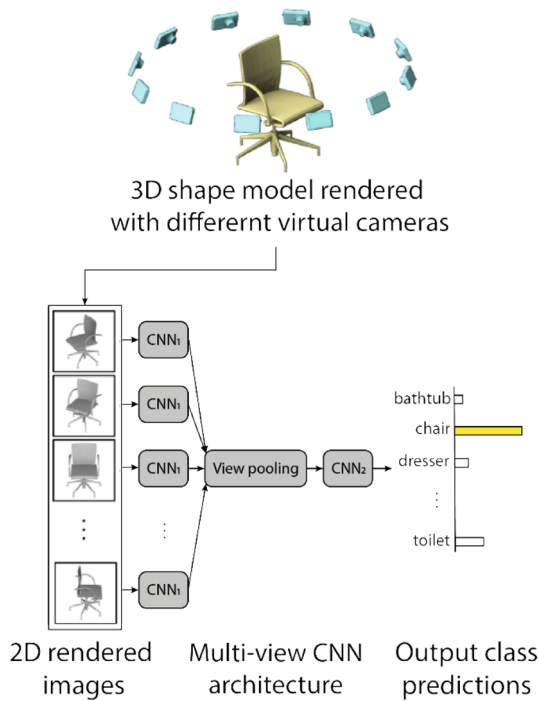


Fig. 4 Model of the multi-view approach for 3D object analyses according to SU et al. [19]

allows more complex insights between the individual images and incorporates them into the classification. The neural net learns to ignore certain views of lower relevance while focusing on more informative perspectives [19]. SEELAND and MÁDER examined different strategies for bundling the information of the individual perspectives in the view pooling layer [20]. The best performance was achieved by the so-called *late fusion approach* in combination with *maximum value fusion* with accuracies between 94–96% for all tested data sets [20].

Object and feature detection

Metrics. The following paragraphs introduce metrics for evaluating and analyzing the performance of object recognition models. *True Positive* (TP) refers to objects that are relevant ($\hat{=}$ meant to be identified) and correctly recognized. *False Positive* (FP) refers to objects that were recognized although they do not exist or that were detected in the wrong position. *False Negative* (FN) describes relevant objects that were not recognized. The metrics do not consider the *True Negative* (TN) predictions, because of the unlimited number of these *non-markings* within one image (see Fig. 5) [23].

These parameters are used to calculate the metrics *Precision* and *Recall*. *Precision* describes how many of the detected objects are relevant [23].

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

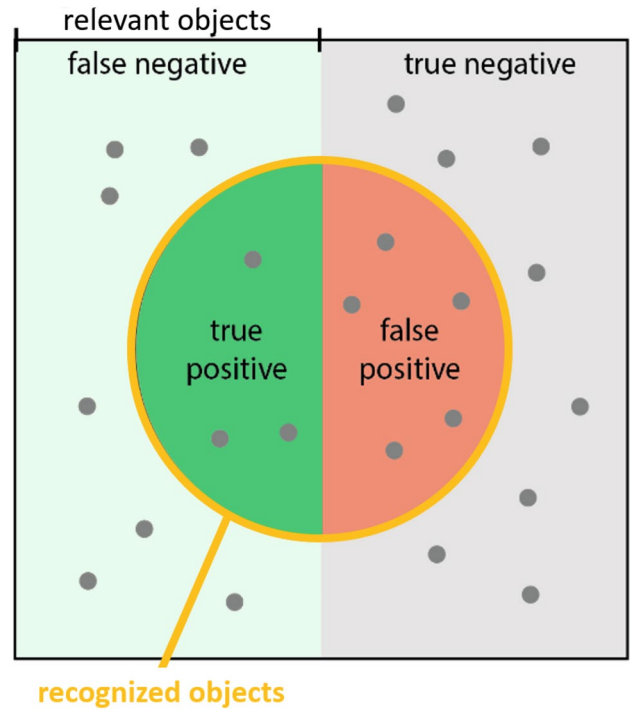


Fig. 5 Illustration of the metrics Precision and Recall

Recall describes how many of the relevant objects were recognized [15].

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

Another metric, the *Intersection over Union* (IoU), indicates the quality of the detections. It measures the area of overlap between the indicated bounding box A (in red) and the true bounding box B (in green) of an object, divided by the union of both (see Eq. 3 and Fig. 6) [23].

$$IoU = \frac{A \cap B}{A \cup B} \tag{3}$$

The area below the graphical plot of *Precision P* over *Recall R* also represents an important metric to assess object detection models and is called *Average Precision* (AP). A large area indicates high *Precision P* and high *Recall R*

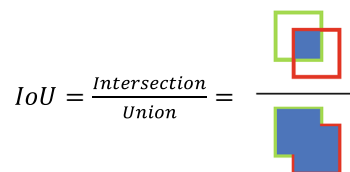


Fig. 6 Illustration of the metric intersection over union based on Padilla et al. (2020, p. 238) [23]

[23]. An exemplary plot can be found within the application example in Sect. 4.2 (see Fig. 30).

$$AP = \int_0^1 P(R)dR \tag{4}$$

The *mean Average Precision* (mAP) results from the mean value of the AP over all classes N [23].

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i \tag{5}$$

Furthermore, the values $mAP^{val}0.5$ and $mAP^{val}0.5:0.95$ are relevant. The value $mAP^{val}0.5$ describes the mean average precision where correctly recognized objects must have a minimum *IoU* value of 0.5. Additionally, $mAP^{val}0.5:0.95$ calculates the *mAP* starting at a required *IoU* of 0.5, then increasing the *IoU* step by step (one step $\hat{=}$ 0.05) up to *IoU*=0.95, hence getting stricter regarding a correct object detection since the overlap must meet the increasing *IoU*.

General approaches. In addition to the classification task, the exact position of objects within a picture or a 3D figure can also be determined. *Object detection* is used, e.g. in the fields of autonomous driving, computer vision, image restoration, robotics and many more [15]. Usually, there are only a few different objects, but they occur more than once and in different shapes or sizes [8].

The so-called *shape segmentation of 3D objects* is one method to consider. In general, shape segmentation enables for the processing of a broader and more difficult input by splitting and classifying objects into *segments*. It can efficiently learn and predict mixed shape datasets, resulting in good segmentation outcomes while simplifying and speeding up learning and inference [24]. Convolutional networks have excelled in a variety of image or object processing tasks, including image classification and semantic segmentation [25, 26]. Kalogerakis et al. proposed a deep architecture for segmenting three-dimensional objects into their semantically identified pieces. To produce coherent segmentations of 3D forms, the method blends image-based fully convolutional networks (FCNs) with surface-based conditional random fields (CRFs) [27]. This method significantly outperforms existing state-of-the-art methods regarding the segmentation benchmark *ShapeNet*. GUAN et al. applied a different technique, interpreting the shape segmentation issue as a point labeling task. An object’s mesh structure is first converted into a series of data points with barycenter and normal vector. The data can be segregated and labeled to find characteristics with trained convolutional models [28].

Object recognition by the identification of so-called *geometrical primitives* is another related topic to mention. Geometric primitives are used to bridge the gap between low-level digitized 3D data and high-level structural information

on the underlying 3D shapes by fitting them to 3D point cloud data. RANSAC-based approaches (random sample consensus) have been considered the standard for fitting problems, but they require careful parameter tuning and thus do not scale well for large datasets with diverse shapes [29]. LI et al. proposed a supervised primitive fitting network (SPFN), an end-to-end neural network that can reliably recognize a variable number of primitives at various scales. Instead of directly predicting the primitives, this method predicts per-point attributes first, then computes the primitive type and parameters using a differential model [29]. Another method involves using a fully convolutional neural network to partition the input point cloud into numerous classes. As a result, segments can be used as primitive hypotheses. Finally, all hypotheses are subjected to geometric verification to correct any misclassifications [30].

Regarding image-based methods, object detection indicates objects and their position via a *bounding box* and a percentage value of the associated class [31]. The YOLO network was first presented in 2015 and continuously improved with five subsequent versions v_i [32]. The approach describes a unified model that represents the input by a single CNN. It focuses on image-based approaches but also works for object detection in videos [33]. The YOLO network calculates all features of the image and indicates all objects simultaneously, explaining the name of the approach: “You Only Look Once”. The procedure for object recognition of the first version of the approach (YOLOv1) is depicted in Fig. 7. In step II, objects within the picture are marked with bounding boxes (II a) and assigned to defined classes (II b). Step III merges the gained insights and generates labeled bounding boxes within the picture.

The YOLOv4 approach compared different network architectures (e.g., *VGG*, *ResNet*) with each other. Shortly after the release of the fourth version, the YOLOv5 version

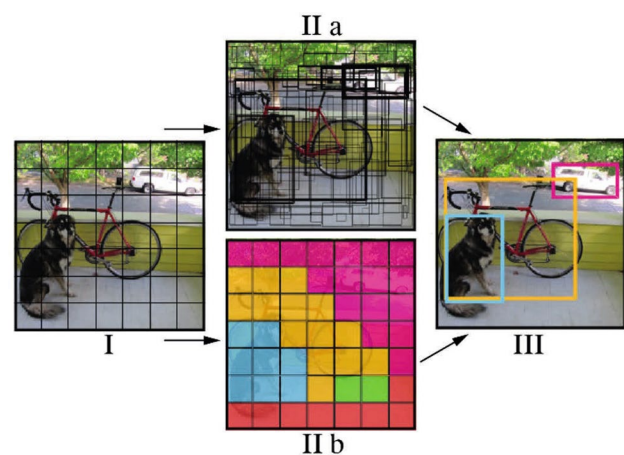


Fig. 7 Illustration of the YOLOv1 approach for object recognition according to Redmon et al. [32]

was presented with minor changes [34, 35]. The YOLOv5 model allows for fast analysis of individual images. In the context of this article, it thus allows for an analysis of up to 3000 images at a time. The YOLOv6 model is accessible as an open source framework [36].

Automatic detection of component features. Current research regarding the automatic detection of component features distinguishes between rule-based and learning-based methods [37]. Rule-based approaches have some disadvantages since the search algorithms are computationally intensive due to the high number of rules that must be implemented for recognizing features [37].

Zhang et al. presented a novel, learning-based method (*FeatureNet*), which uses 3D CNNs to recognize features from CAD models of mechanical components [38]. In contrast to rule-based approaches, there was no need for implementing specific rules per feature. To achieve shorter computing times, Shi et al. (2020) used a learning-based approach based on the multi-view model to analyze 2D pictures [37]. In a proceeding article, SHI et al. suggest an approach based on the *Single Shot Multibox Detector* (SSD) architecture that three-dimensionally localizes features after recognizing them in 2D images [37, 39].

2.3 Research gap

According to Sect. 2.2, many research activities have been carried out in the field of picture classification and object recognition. These methods generally enable a classification based on 2D image data. However, it is currently not possible to conduct automatic, geometric analyses of components while evaluating respective technology alternatives early in product development. Furthermore, current object recognition approaches often focus on identifying imperfect products after their production [40–42]. The article thus aims at answering the following research questions:

- Can picture classification and object recognition models be used for the automatic screening of metallic components to identify suitable manufacturing technologies?
- Are these AI-based classification and object detection models able to analyze different manufacturing technologies within one multi-class approach?
- Can 2D image analyses accelerate geometric analyses of many 3D components at a time compared to CAD-based approaches?

3 The geometry module: general methodology for geometric analyses on a component level

This section introduces the general approach for analyzing components' geometries based on component images. Section 3.1 presents the approach for the classification of components by manufacturing technologies. Section 3.2 focuses on the object detection within the components to generate additional data and support the classification.

It should be noted that the geometry module focuses on the analysis of existing modular car body structures and the respective components. Geometric analyses of potential new components such as casting knots (so-called integral components) that result from merging several smaller parts require a different approach since there are no pictures of these fictional new components.

3.1 Picture recognition and classification

The picture recognition and classification method aims at distinguishing component pictures by different manufacturing technologies and follows the approach depicted in Fig. 8.

Step 1 and 2: Retrieving and labeling image data.

Before the classification models are built, trained, and validated, an image database must be created. CAD compatible methods such as VBA macros allow for the automatic derivation of e.g., seven pictures per component within the structure of interest in programs, such as CATIA V5 R29 (Dassault Systèmes, France). The pictures can be created based on components that are either already initially generated in current development projects or based on preceding product generations. These seven perspectives represent the main views of a CAD program: A top, bottom, rear, and front view, a view from both sides, and an isometric view.

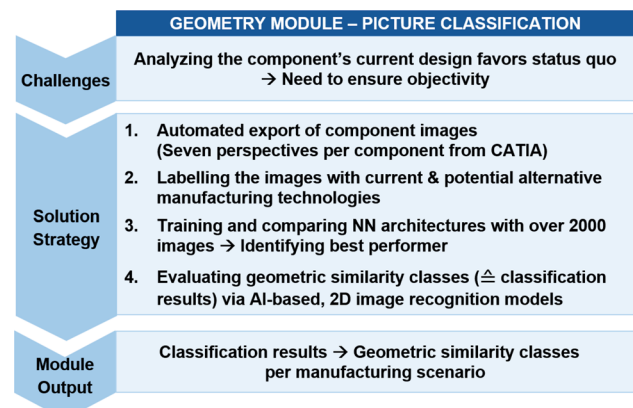


Fig. 8 Approach for picture classification within the geometry module

To create the views, auxiliary information, coordinate systems, and the structure tree are hidden in the CAD interface. Furthermore, there is no zoom on the components to ensure consistent proportions between smaller and larger components. Following the image export, the images should be tailored to reduce the file size and to ensure a consistent ratio of, e.g. 450×450 pixels. This step can be automatized by a Python script.

After retrieving the images, each component and the respective images are labeled with their current manufacturing technology of previous product development projects and potential alternative technologies. These alternative technologies can be identified by interviewing manufacturing experts. The interviews thus must be conducted on single time before the training phase. Based on this, different AI approaches can be trained and compared for a geometric analysis that supports the manufacturing technology selection.

Step 3: Classification approaches. This section introduces different approaches for analyzing and classifying pictures that are subsequently tested in Sect. 4.1 to identify the best performing model. The competing approaches are: *Stitching*, *7 Neural Networks*, *Single CNN with sum score fusion*, and *Multi-View*.

Stitching. The stitching approach aligns all 2D images in a row and merges them into one large image (see Fig. 9). These images are then used to train the AI classification model.

Pre-trained networks cannot be used due to the non-square ratio of the composite images since available pre-trained architectures are tailored for squared images. A grid search led to the following architecture (see Fig. 10) that was used for the training phase:

- sequential model
- each level with one input and one output tensor

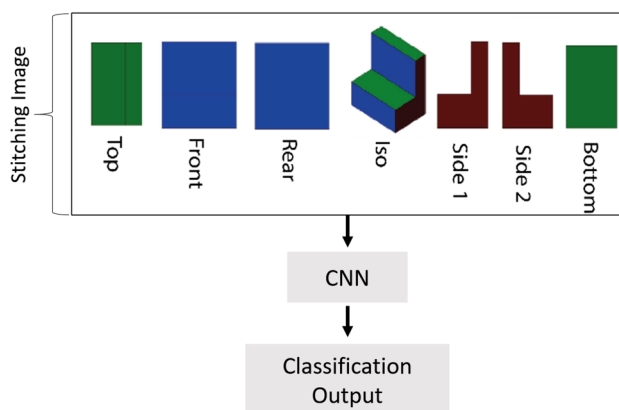


Fig. 9 Exemplary stitching image after merging all perspectives and the schematic stitching approach

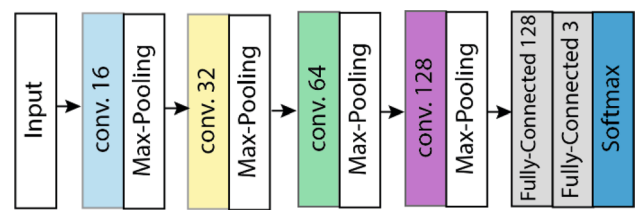


Fig. 10 Schematic representation of the model architecture of the stitching approach

- four convolutional layers (ReLU activation function)
- each layer followed by a max-pooling layer to reduce the dimensional complexity
- fully connected layer at the end with 128 neurons and a ReLU activation function.

7 neural networks (7NN). The 7NN approach sets up a separate network for each of the seven perspectives. This way, certain characteristics of the different manufacturing technologies might be easier to recognize in individual views. The classification outputs of each view are first summed up (sum score fusion), followed by a mean value calculation (see Fig. 11). Furthermore, it is possible to weigh individual results of the views, potentially leading to better predictive accuracy for the entire component. A grid search identified an architecture that is similar to the stitching model of Fig. 10 except of one difference: the 128-convolutional layer has been removed.

Single CNN with sum score fusion. This section describes an approach that uses a pretrained single CNN (VGG16 architecture) to manage the seven perspectives. The classification values are then averaged by the *Sum Score Fusion* method, as illustrated in Fig. 12.

Grid testing led to two fully connected layers with 1024 neurons, the first was combined with a 50% dropout layer. Another fully connected layer with three output neurons (*Softmax* activation function) represents the end to generate the percentage values for the three classes (see Fig. 13).

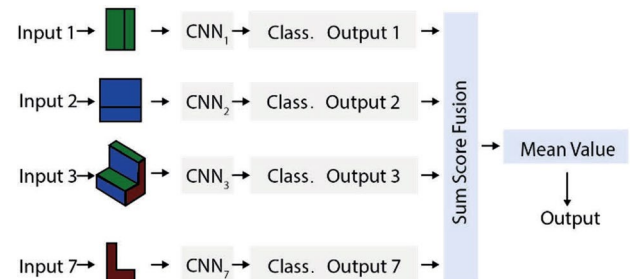


Fig. 11 Schematic representation of the merged classification results of the individual networks via a sum score fusion

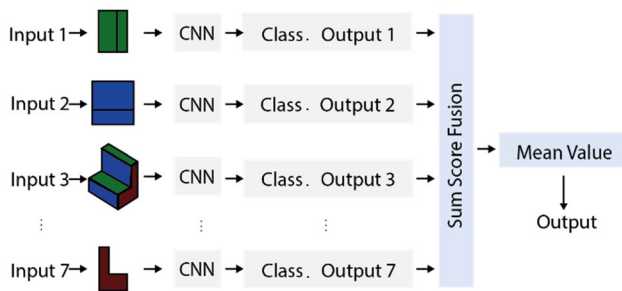


Fig. 12 Schematic representation of the single CNN approach, merging the classifications of the individual views

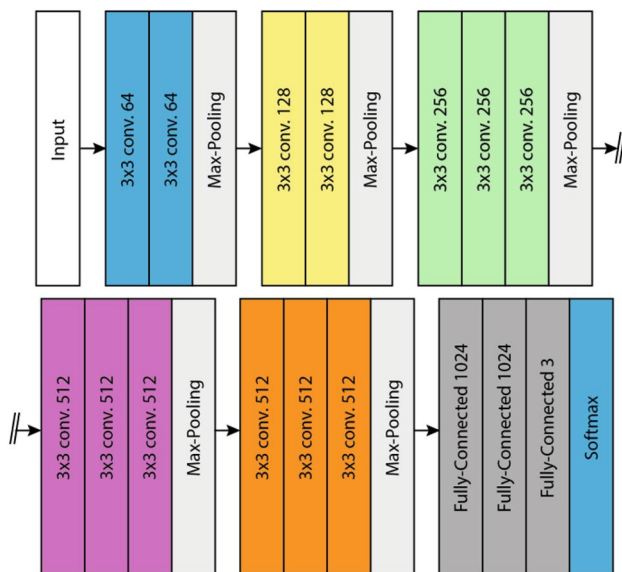


Fig. 13 Schematic representation of the model architecture of the single CNN approach

Multi-view. This approach links the characteristics of the previous variants. It connects the 2D image information of the individual views to get further insights into the 3D component characteristics. To achieve this, the model follows the approach according to Su et al. [19]. A single CNN is used to analyze all component perspectives. Different versions and architectures are compared with each other: ResNet34, ResNet50 [21], and VGG16 [17]. In a subsequent step, insights across all views are bundled within a view pooling layer using a late fusion with a max value function and analyzed by a second CNN, as illustrated in Fig. 14 [19, 20].

Step 4: Module output. Two measures were implemented to mitigate the bias towards the technological status quo of the components:

Measure 1. In training, the neural net first learns how components of different manufacturing technologies look

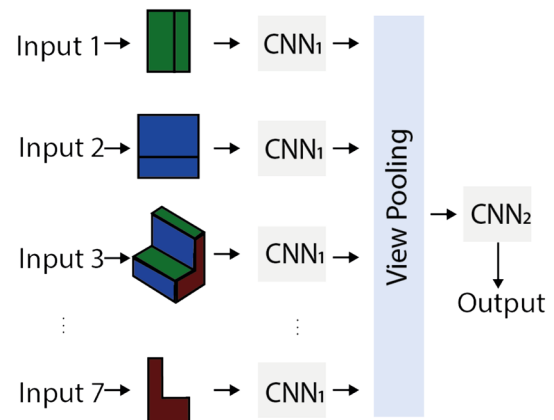


Fig. 14 Schematic representation of the model architecture of the multi-view approach, based on SU et al. [19]

like. However, the neural net must learn to identify characteristics of more than one manufacturing technology in the components' pictures instead of reinforcing the exclusive recognition of characteristics of the current technology. The pictures are thus not only labeled with the current manufacturing technology but also with potential alternate technologies (so-called double or triple labeled pictures). Figure 15 illustrates the seven perspectives of an exemplary component. Since deep drawing represents the current technology of this part, the classification model identified mostly characteristics of previously seen deep drawing components. However, it also indicates pressing potential. These two manufacturing technologies might be recognized by the AI model due to the elongated character of the component (pressing characteristic) and the U-profile in two of the perspectives (deep drawing indicator). The goal of identifying characteristics of more than just the current technology was thus met.

Measure 2. The geometry module generates geometric similarity values ($\hat{\Delta}$ classification results in percent) regarding the technology alternatives. This percentage-based classification approach was chosen over a binary classification logic (clear indication of one class with the value 1, all other classes 0) to equip the geometry module with the ability to identify more than just one class per component. When applying the selected model on unlabeled image data, the resulting classification results and similarity values represent the necessary effort to redesign the respective component towards a different manufacturing technology. The higher the classification value, the more geometric characteristics of the respective technology were found by the underlying AI model. Consistently, low similarity values indicate that the component does not embody a lot of characteristics of the respective technology.

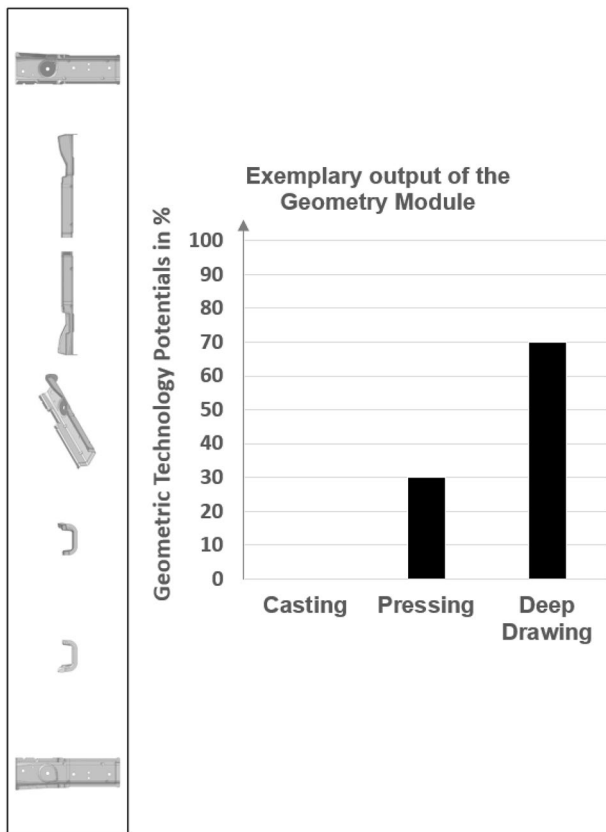


Fig. 15 The seven perspectives of an exemplary component (l.) and the respective geometric classification results (r.)

3.2 Object detection

The object detection aims at the identification of component features within the pictures and follows the approach depicted in Fig. 16. This approach was chosen over a simple data extraction out of STEP files, as well as over the

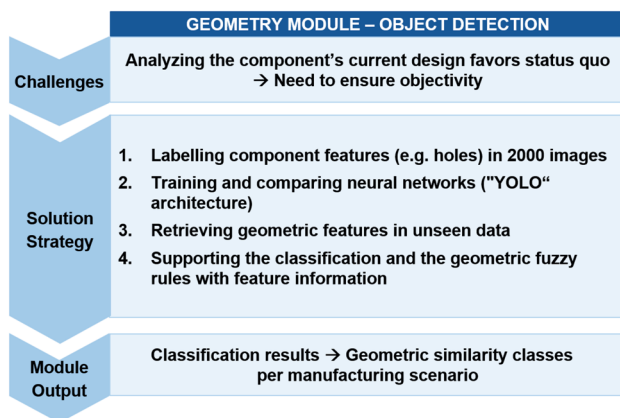


Fig. 16 Object detection approach within the geometry module

application of 3D-based shape segmentation and geometrical primitives because of these reasons:

1. Only some of the features of interest (holes, coinages, flanges, closed profiles) are documented in STEP files. However, the chosen approach must be able to identify all features of interest. Furthermore, the geometric data, e.g. regarding the number of holes, is often not consistently generated by the responsible CAD designer (e.g. manual vs. automatic creation and naming of holes), causing strong challenges for automatization.
2. Due to the close connection of the classification and the object detection task within the geometry module, the already existing component pictures (out of the classification approach) can be used to ensure a lean and clear data flow within the module. This strongly favors the usage of existing component images for a highly automated object detection.
3. The intended short computing times of the geometry module must be maintained even for the analyses of hundreds of components (= thousands of pictures) within one screening run. Based on existing 3D-based approaches, e.g. for similarity assessments in CATIA, these short computing times cannot be reached with RAM-intensive, 3D approaches that must open, segment and analyze every CAD file.
4. The accuracy of the image-based object detection approach does not have to meet perfection due to the mitigation of tolerable inaccuracies (compared to 3D-based approaches) by the fuzzy systems and its binary rules in the overall methodology. More importantly, the achieved precision of this article's object detection approach meets the required accuracy whilst ensuring short computing times (see Sect. 4.2).

The detected features are relevant for supporting the classification approach of Sect. 3.1.

Step 1: Building the image database. The 2D image data of the classification task consists of component images that were designed for different manufacturing technologies. The graphic software tool *LabelImg* [35] can be used to mark and label individual component features (e.g., holes or coinages) within the images. The features were translated into a unique numerical identifier. The X and Y coordinates of the markings' center points within the picture as well as the width and the height of the rectangular marking were automatically documented.

Step 2 and 3: Object detection approach. Due to its modern architecture and outstanding performance, the approach focuses on the *YOLOv5* model that follows a *one-stage* approach of object detection [34]. The *YOLOv5* architecture can be trained in four different model sizes (S, M, L, XL) that differ in the number of layers. The model sizes

embody a compromise between recognition accuracy and processing speed. In the scope of this article, it is important to choose a model that guarantees a short processing time for a large number of images and components. The models were pretrained based on the *COCO database*, an extensive image collection of all kinds of pictures [34]. The training results and achieved accuracies can be found within the application example in Sect. 4.2.

Step 4: Providing the geometric fuzzy rules with additional data. The features identified by the object detection model can be looped back into the data flow and used as additional input for the fuzzy rules. For instance, closed profiles can now be detected within components and indicate high pressing potential (see Sect. 4.3 for more details).

3.3 Part screening: analyzing the module's outputs

The geometry module contributes to the overall screening approach by enabling the addition of further fuzzy rules for each manufacturing technology. These fuzzy rules are based on the geometry module's outputs. The output is the classification result (geometric similarity classes) for each component and technology scenario, supplemented by recognized features within the pictures of the components. The extended rule sets hence consist of rules out of all four screening modules and ensure a holistic manufacturing technology comparison. Exemplary fuzzy rules can be found in Sect. 4.3.

4 The geometry module: exemplary application in car body development

The *Geometry Module* was applied and evaluated using the automobile sector as an example. The intricate construction and design of car body components are dependent on the selection of the appropriate manufacturing method for each part at an early stage. So far, this selection relied on human experience and non-automatic evaluations. It is thus skewed in favor of previous product development projects and chosen manufacturing technologies. The geometry module addresses this problem by objectifying geometric characteristics to reduce human bias and the preference of the status quo. It thus represents an important source of information for the overarching screening methodology as part of the car body development and production process (see Fig. 17).

4.1 Picture recognition and classification

Section 4.1 applies the general approach of Sect. 3.1 and tests its different models to identify the best performer.

Retrieving image data. A total of 1487 components were labeled and added to the database: 76 casting, 180 pressing, and 1231 deep-drawn components across different vehicle

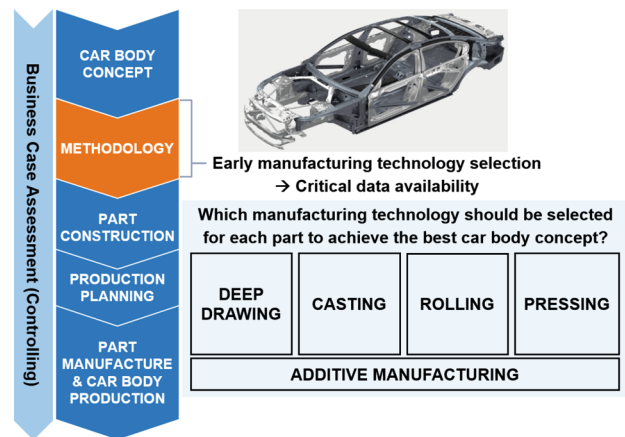


Fig. 17 The geometry module supplements the main author's screening methodology (orange) [4] as a part of the car body development and production process (color figure online)

models. Black and white illustrations were used to mitigate the influence of colored component surfaces. It must be noted that casting or pressing components are underrepresented in modern car body architectures. An imbalance between classes is a common problem in real-world applications. This problem affects the performance of classification models since the minority class often is overlooked [43]. One countermeasure is to use a reduced amount of deep drawing components in training to compensate for the imbalance. Another is to apply cost-sensitive learning to solve this issue by enforcing the learning effects of underrepresented classes. Misclassification costs are introduced to punish wrong classifications differently depending on the class size [43].

Classification approaches.

This paragraph compares the different classification approaches regarding their performance in training, validation, and testing. To distribute the data volume for each class as evenly as possible, all approaches were trained and validated with a data base of 532 pictures of casting components, 833 pictures of pressing components, and 833 pictures of deep drawing components. This data base was split into 80% training and 20% validation data. After the validation, the trained models were applied on additional unseen test data (around 300 car body components) to create and quantify truth matrices. This test data was retrieved out of another vehicle model that was not part of the training and validation data base. The optimal parameters (number of epochs, batch size, learning rate etc.) were identified with a grid search. Table 1 lists the different models and their parameters. The best performing approach (highlighted in green) was chosen for further development based on the initial comparison.

Stitching. Figure 26 (see Appendix) illustrates the performance of the best stitching version. It indicates a strong

Table 1 Overview of the classification approaches and their achieved validation accuracy (color figure online)

Classification approaches	Training epochs	Optimizer	Learning rate	Achieved validation accuracy
<i>Stitching</i>	25	Adam Optimizer	0.001	64 %
<i>7 Neural Networks</i>	25	Adam Optimizer	0.001	67 %
<i>Single CNN</i>	600	SGD Optimizer	0.00001	80 %
<i>Multi-view</i>	60	SGD Optimizer	0.0005 & 0.00001	87 %

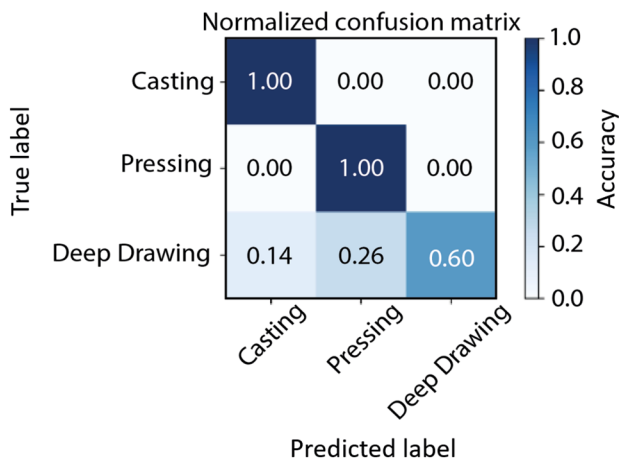


Fig. 18 Truth matrix for evaluating the performance of the stitching approach on around 300 unseen components (test data)

learning effect from epoch 2 to 5 and overfitting behavior from epoch 12 onwards with an increasing loss of validation. This behavior is triggered by the large image dimensions, as this approach cannot rely on pre-trained weights. It thus has to learn small features such as edges or curvatures as well as more detailed component characteristics. The poor validation accuracy might be caused by the lower number of training data points compared to other approaches, as 7 pictures are merged to one.

The trained model was then applied to selected test data of another vehicle model, Fig. 18 illustrates the evaluation in form of a normalized truth matrix. The matrix shows that

deep-drawn components cannot be clearly distinguished from the other classes. This approach can thus be rejected.

7 neural networks. The 7NN-models were trained for each of the seven perspectives. Furthermore, a 20% dropout level turned out to increase the performance and was inserted before the fully connected layer. Figure 27 shows the training results of the perspective “bottom”.

The plots show hardly any overfitting since the courses of training and validation proceed on a similar level in terms of accuracy and loss. This can possibly be attributed to the seven individual networks that mitigate overfitting regarding the overall classification. Further epochs would not improve the performance since the curves are almost constant from epoch 20 onwards. The model shows a maximum accuracy of 74% in training and 67% in validation. Table 2 shows the results of the seven networks when applied to unseen part data. The deep drawing performance strongly influences the overall performance for *all classes* due to the overrepresentation of deep drawing parts in the unseen data.

Merging the classification results of the seven networks based on the maximum class values leads to the following confusion matrix, see Fig. 19. The truth matrix indicates difficulties with the classification of deep-drawing components due to a poor forecast accuracy for the top and side views. This issue was addressed by specific perspective weightings that did not lead to significantly better results. For this reason, the approach was discarded.

Single CNN with sum score fusion. The courses of the training and validation accuracies show no signs of overfitting for the single CNN approach. Although the curves diverge after around 200 epochs, they also increase the validation accuracy in training (see Fig. 28).

The evaluation of the trained models based on selected, unseen test data led to a maximum accuracy of 80% across all classes. Out of 181 components, 25 were incorrectly classified. Figure 20 shows the results of the best version in form of a confusion matrix. Overall, the single CNN approach with pre-trained weights led to a better performance.

Multi-view. Different multi-view versions were trained and compared. The VGG16 architecture was used for all versions with and without pre-trained weights. They were trained for 40 epochs in phase 1 and a learning rate of $lr=0.0005$. Due to the promising performance, the best

Table 2 Overview of the evaluation of the accuracy of the seven networks for unseen component data

	Top (%)	Front (%)	Rear (%)	Iso (%)	Side 1 (%)	Side 2 (%)	Bottom (%)
Casting	100	100	100	100	50	100	100
Pressing	100	83.3	33.3	100	100	66.7	100
Deep drawing	43.7	58.1	62.6	71.4	50	57.5	64.4
All classes	46.2	59.3	62.1	72.5	51.7	58.2	65.9

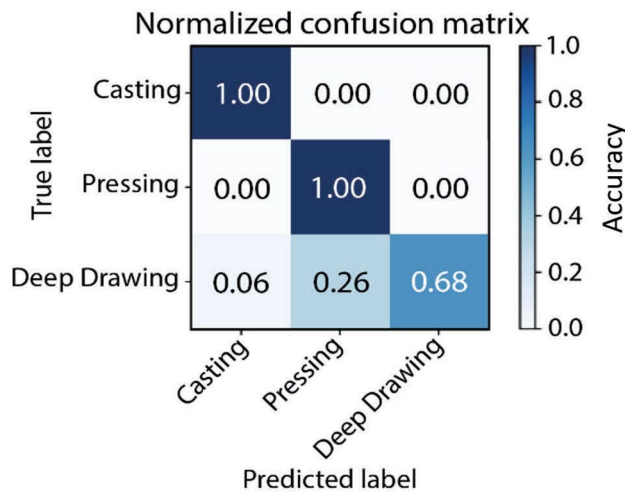


Fig. 19 Truth matrix for evaluating the 7NN approach on around 300 unseen components after averaging the seven individual outputs

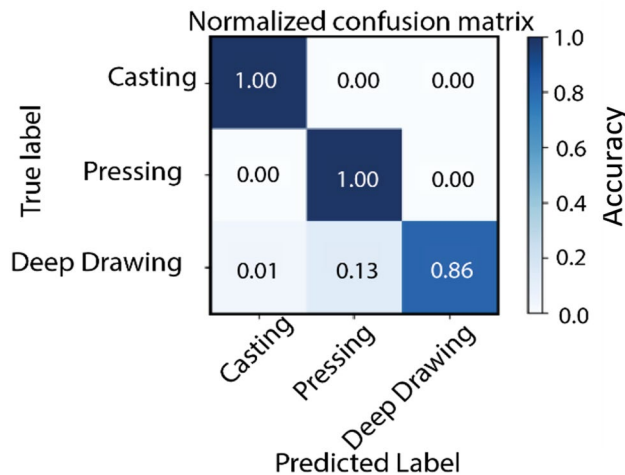


Fig. 20 Truth matrix for evaluating the performance of the best version of the single CNN approach on around 300 unseen components

version was trained for further 20 epochs in phase 2 with a learning rate of $lr=0.00001$. The best multi-view version achieved the highest accuracies compared to all other approaches. Precisely, it reached a validation accuracy of 87% and was thus chosen for further development and improvement.

Evolution of the multi-view approach.

After identifying the multi-view approach as the best performer amongst all compared models (see previous paragraphs), two evolutions were tested:

First: Evolution of the multi-view approach through a data base extension.

Second: Evolution of the multi-view approach through an additional class extension.

Multi-view with data base extension. First, the data base was extended to 518 casting, 1246 pressing, and 7518 deep-drawing pictures. The increase of deep-drawn components needed to be compensated by cost-sensitive learning and its emphasis on learning effects of underrepresented classes. The additional training data for the deep drawing class was expected to improve the accuracy of the often incorrectly classified deep drawing parts that represented the dominating technology in modern car body architectures. The VGG16 architecture remained for all versions with and without pre-trained weights. It was trained for 40 epochs in phase 1 using the SGD optimizer and a learning rate of $lr=0.0005$. In phase 2, it was reduced to $lr=0.00001$ and trained for further 20 epochs.

Class-specific weights of the cost-sensitive approach led to higher accuracies. The best version achieved a validation accuracy of 98.9%. The influence of the lower learning rate can be seen in the progression of accuracy and decrease of loss after epoch 40 (see Fig. 29).

Multi-view with class extension. As the next step towards the Geometry Module's use case, the classes were extended with two further labels (see next paragraph). This measure avoids an exclusive confirmation of the components' current manufacturing technologies (= the ones in the images) and alternative technology characteristics to be overseen. The *casting/deep drawing* class represents components, which can be geometrically produced as casting as well as deep-drawing components. The *pressing/deep drawing* class contains components that can be manufactured as pressing or deep drawing parts. After labeling the database, the training data contained 54 casting, 87 casting/deep-drawing, 131 pressing, 145 pressing/deep-drawing, and 902 deep-drawing components. Pre-trained weights were used for all versions in which a VGG16 architecture was compared with a ResNet50 architecture.

Class-specific weights were used in training of the best performing version. Phase 1 consisted of over 70 epochs, an SGD optimizer, and a learning rate of $lr=0.00003$. Phase 2 added further 40 epochs and applied a learning rate of $lr=0.00001$. Phase 1 focused on training the second CNN after the view pooling layer while freezing the first CNN with pre-trained weights to shorten training times (model architecture see Fig. 14). The fine tuning in phase 2 of the first CNN showed more effect in the ResNet50 architecture than in the VGG16 architecture depicted below. However, fine tuning prevented an increasing validation loss after 70 epochs. The accuracy of the validation curve with class-specific weights is well below the validation curve without weighting (see Fig. 30). However, the weighted version showed better generalization when applied to unseen data after the validation phase. Furthermore, it led to more correct double-label predictions (see

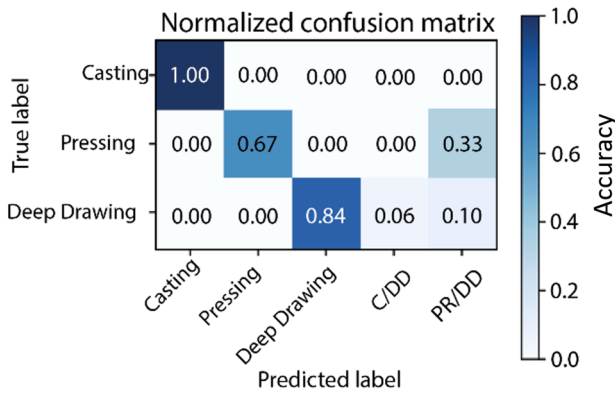


Fig. 21 Truth matrix of the best multi-view version with class extension applied on around 300 unseen components (C: casting, PR: pressing, DD: deep drawing)

Fig. 21). These double-labels are wanted since they offer insights on potential changes of manufacturing technologies. Overall, these arguments overcompensate the lower validation accuracy of the weighted approach that is hence considered to be the best performer (VGG16). The relatively high loss at the end of training can be explained by the small amount of training data for the individual classes due to the data distribution over five instead of three classes.

At first sight, the best performer showed lower accuracies for unseen deep-drawn components with 16% of false classifications (see Fig. 21). However, these wrong assignments occur exclusively in the double-labeled classes. These components were assessed with technology experts and considered to be correctly classified since they showed characteristics of both indicated technologies. Hence, these components can be manufactured in more than one technology without the need for extensive redesign. In fact, these classifications indicate that the respective components can be manufactured in two different technologies. These suggestions were confirmed by interviews with construction experts of the different manufacturing technologies.

Table 3 lists the class-specific weights of the best-performing version. Higher weights indicate a harder punishment for wrong classifications and a higher reward for correct classifications. Underrepresented classes such as casting were attributed with higher weights.

Table 3 Overview of the class-specific weights of the multi-view approach with class extension

	Casting	Pressing	Deep drawing	Casting/ deep drawing	Pressing/ deep drawing
Weight w_i	4.071	1.678	0.244	2.527	1.156

Best performer. Based on the presented parameters, the multi-view with class extension is the best performing version. Components with close classification results, e.g., when two manufacturing technologies *compete*, are of particular interest within the screening tool as they show potential for a change in manufacturing technology.

4.2 Object detection

The upcoming paragraphs apply and analyze the general approach of Sect. 3.2 regarding its performance.

Building the image and feature database. The following component features are relevant in the use case and were first marked within the pictures to enable the subsequent training of the neural net (see Fig. 22).

Hole/bore. Different shapes of holes were considered and labeled within the pictures. In contrast to the feature *closed profile*, holes are mostly surrounded by a thick layer of component material (> 5 mm).

Closed profile. Closed profiles are surrounded by only a thin layer of component material (max. 3 mm). Furthermore, a feature called *closed profile not white* was introduced to also mark closed profiles without a white background (e.g., due to visible underlying component material depending on the perspective).

Coinage. Coinages occur in the form of depressions or notches and were also marked within the pictures.

Curvature. Curvatures have been marked to grasp the complexity and the three-dimensional character of a component.

Flange. To anticipate possible additional expenses by folding or bending operations, flanges and bent component parts were marked.

The number of all marked features per class is shown in Table 4.

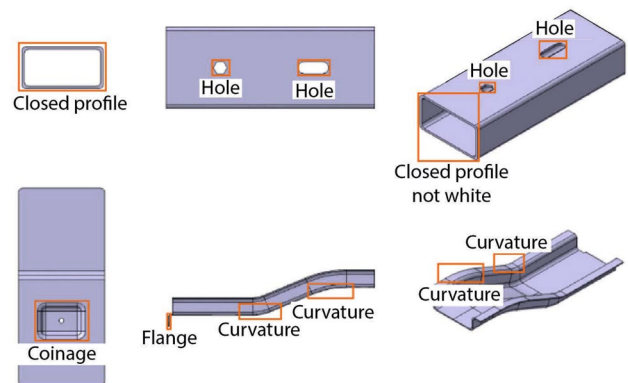


Fig. 22 Illustration of the different component features using exemplary components, with indicated markings and associated class names

Table 4 Overview of the number of marked features per class within the training data

	Hole/bore	Closed profile	Closed profile not white	Coinage	Curvature	Flange
Number of features	11,160	260	2948	2948	976	548

Table 5 Overview of the different network architectures of the YOLOv5 model

	S-Net	M-Net	L-Net	XL-Net
Quantity parameter (in million)	7.2	21.2	46.5	86.7
CPU interference time (in ms)	98	224	430	766
mAP ^{val} 0.5	56.0	63.9	67.2	68.9
mAP ^{val} 0.5:0.95	37.2	45.2	48.8	50.7

Object detection approach. The labeled data was divided into 80% training and 20% validation data. YOLO networks of the sizes S, M, and L were trained in two different variants. One variant used pre-trained weights, the other randomly initiated weights. The training parameters and results are shown in Table 5 using the mean average precision *mAP*.

A higher resolution of the images enables better recognition of smaller objects [42]. The images in the database have a resolution of almost 1000 × 1000 pixels and were used for the training of the S network. 864 × 864 pixels images were used for training the M and L networks to avoid overloading of the server that used three *GeForce GTX TITAN X* GPUs with twelve GB of RAM (Nvidia Corporation, Santa Clara, United States). For the versions with pre-trained weights (V1, V3, V5), training lasted from 400 to 500 epochs, whereas the versions without pre-trained weights (V2, V4, V6) were trained throughout 800 epochs. The lower number of epochs for versions V1, V3, and V5 was chosen to prevent overfitting.

Based on Table 6, the use of pre-trained weights shows a positive influence on the accuracy of smaller networks. However, there is hardly any difference in the *mAP* values for the versions V5 and V6 of the L net. The mAP^{val}0.5:0.95 values of the different versions accumulate in a relatively narrow range from 0.443 to 0.477, which is why

no version can be excluded or preferred. The larger input dimensions of versions V1 and V2 enable better detection of the smaller bores, which is reflected in the best mAP^{val}0.5:0.95 values of 0.511 (V2) within the *holes/bores* class.

Figure 23 illustrates the development of Precision over Recall of the best performing version V2 across all classes during training. It is apparent that the object recognition provides very good results for *holes/bores* and *closed profiles* with a precision of over 90%. These features are of particular interest as they strongly support the overall methodology and its fuzzy rules. The achieved precision of over 90% is sufficient since the fuzzy rules only require a binary input (e.g. *closed profiles* yes/no; see Sect. 4.4). Furthermore, the curves indicate increasing Recall at a constantly high Precision level. In contrast, the plot shows a poorer performance for curvatures, possibly because labeled curvatures often strongly differ from each other in terms of shape, orientation, and size. Overall, the dark blue curve for all classes is in an acceptable range.

The knowledge regarding components' features can be used as additional input for the fuzzy rules (e.g. closed

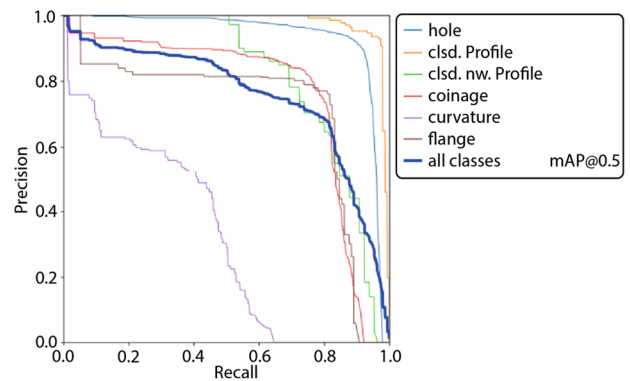


Fig. 23 Course of Precision over Recall for the different component features during the training of version V2

Table 6 Validation results of the versions of the YOLOv5 architecture

Version	Grid	Pretrained weights	Image size	Precision	Recall	mAP ^{val} 0.5	mAP ^{val} 0.5:0.95
V1	S	Yes	1024 × 1024	0.847	0.731	0.748	0.471
V2	S	No	928 × 928	0.768	0.765	0.754	0.443
V3	M	Yes	864 × 864	0.857	0.727	0.734	0.471
V4	M	No	864 × 864	0.821	0.763	0.761	0.463
V5	L	Yes	832 × 832	0.837	0.747	0.742	0.477
V6	L	No	832 × 832	0.836	0.756	0.766	0.477

profiles for pressing assessments, see Sect. 4.4) and to improve the *Cost Modules*' cost predictions. These adjustments within the *Cost Module* are currently under development.

4.3 Application of the Geometry Module

This section elaborates on the application of the Geometry Module with the multi-view approach as the identified best classification model (Sect. 4.1) and YOLO V2 as the best object detection model (Table 6, Sect. 4.2). It was applied for an SUV with more than 400 car body components. The classification results are shown in Fig. 24 and illustrate the distribution by manufacturing technology.

Deep-drawing components make up the largest share with 81%. This result is plausible due to the dominant use of this manufacturing technology in modern car bodies. Table 7 shows the actual distribution of each manufacturing technology versus the predictions of the Geometry Module. Classifications by the *casting/deep drawing* or *pressing/deep drawing* double class were counted with 0.5% points per individual class. *Unknown* components are parts without clear information regarding the current manufacturing technology, possibly due to a lack of information in the databases. They are considered anyway to ensure an application for the holistic car body architecture of the chosen vehicle model.

Overall, the geometry module's output meets reality while suggesting potential changes in manufacturing technology for around 7–8% of the components. The goal of the geometry module was thus met.

4.4 Integrating the geometric parameters into the part screening methodology

An overview of all fuzzy input variables of all four modules is illustrated in Table 8. The Geometry Module provides two new input features: The *classification* results and the binary value *closed profile* (in red). The classification was

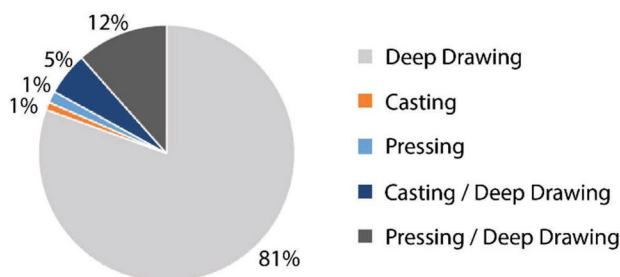


Fig. 24 Evaluation of the results after applying the Geometry Module, distribution by manufacturing technology

Table 7 Comparison of the actual technology distribution and the results of the Geometry Module's prediction

Manufacturing technology	Actual proportion (%)	Predicted proportion (%)
Deep Drawing	89.5	89
Pressing	1.9	7.5
Casting	0	3.5
Unknown	8.6	–

implemented via a trapezoidal membership function in three increments. Closed profiles are indicated by a binary value and processed via a triangle membership function.

Rules of the geometry module.

This section provides an overview of the Geometry Module's fuzzy rules for each manufacturing technology. These rules can be adapted depending on the intended strictness within the use case: the higher the required geometric classification percentage is to reach a *medium* or *high* evaluation, the stricter the technology assessment is.

The following intervals have been chosen for the evaluation of each technology and component (also see Fig. 30):

1. If the maximum classification value of a technology ($\hat{\Delta}$ Output of the Geometry Module) is less than 40%, the respective technology is evaluated with a low potential for the component.
2. If the maximum classification value of a technology ranges between 40 and 60%, the respective technology is evaluated with a medium potential for the component.
3. If the maximum classification value of a technology is more than 60%, the respective technology is evaluated with a high potential for the component.

Deep drawing. Out of the picture classification task, the three classes *deep drawing*, *casting/deep drawing* as well as *pressing/deep drawing* are relevant. The fuzzy rules are

Table 8 Overview of the fuzzy input variables of all four modules, type of membership function, and respective value ranges (color figure online)

Input variable	FT specific	Membership function	
		Type	Value / Interval
Material	no	Singleton function	1; 2
Weighting factor costs	no	Trapezoidal function	0-0.5; 0.5-0.8; 0.8-1
Part volume	no	Trapezoidal function	0-10; 10-50; 50-100000
Corrosion requirements	no	Singleton function	1; 2; 3; 4
Crash requirements	no	Singleton function	1; 2; 3; 4
Temperature requirements	no	Singleton function	1; 2; 3; 4
Lightweight ratio	no	Trapezoidal function	0; 0-0.2; 0.2-1
Part functionality	no	Singleton function	1; 2
Name push	yes	Singleton function	1; 2
Classification	no	Trapezoidal function	0-40; 40-60; 60-100
Evaluation closed profile	yes	Singleton function	1; 2

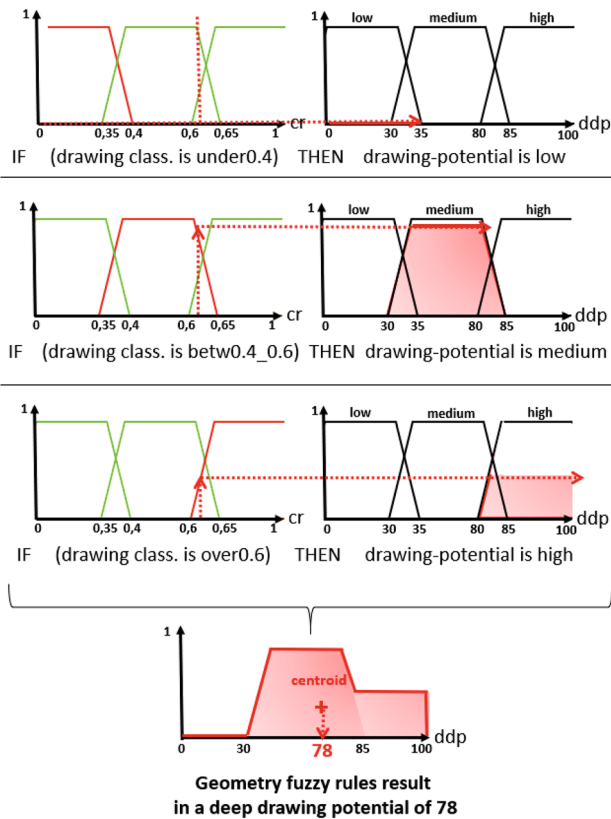


Fig. 25 Visualization of the geometry module's three deep drawing fuzzy rules for an exemplary component with a deep drawing cr of 61%

applied based on the maximum classification value across these three classes (= output of the geometry module).

Figure 25 visualizes the deep drawing scenario of the fuzzy rules for an exemplary component with a classification result (cr) of 61% or 0.61. The selected defuzzification method *centroid* calculates the geometric center of the resulting red areas and thus led to a deep drawing potential (ddp) of 78 for this component.

Pressing. The classes *pressing* and *pressing / deep drawing* are relevant to assess the geometric pressing potential of a component. The fuzzy rules are applied based on the maximum classification value across the two classes.

To realize a distinction between the two extrusion technologies pressing and rolling, the feature *closed profile* was used as follows:

1. If no *closed profiles* are recognized, the component is evaluated with a low pressing potential.
2. If *closed profiles* are recognized, the component is evaluated with a high pressing potential.

These rules embody the characteristics of the pressing technology since pressing is ideal to manufacture extruded components that require closed profiles.

Rolling. Due to the low number of rolling parts in modern car body architectures and the respective low number of rolling images, this extrusion technology relies on the classification results of the extrusion classes *pressing* and *pressing/deep drawing*. Additionally, the feature *closed profile* was used:

1. If no *closed profiles* are recognized, the component is evaluated with a high rolling potential.
2. If *closed profiles* are recognized, the component is evaluated with a low rolling potential.

Casting. The classes *casting* and *casting/deep drawing* are relevant. The fuzzy rules are applied based on the maximum classification value across the two classes.

Threshold check. Additionally, it is checked whether a technology's maximum classification value exceeds a threshold of 20%. If not, the respective technology potential is set to zero since it is not suitable to create the component's required geometry.

Additive manufacturing. There are only a few additively manufactured components in the series production of car bodies. Furthermore, the high design freedom of AM components allows to manufacture any geometry. Hence, the AM analysis requires a different approach and cannot rely on classification results. The component features (recognized by the object recognition) can be used to derive geometric criteria for the evaluation of AM potentials. The following rules have been implemented:

- i. If the maximum edge length of a component exceeds 300 mm, the component is discarded for AM due to the 3D printers' limits in building space and economic disadvantages of AM.
- ii. The ratio of the volume of the component's surrounding bounding box in x, y, and z-direction over the component's volume itself is relevant. The higher this ratio, the less space is occupied by the component within its surrounding bounding box. A high ratio indicates a strongly curved component with a complex geometry. This complexity causes high conventional manufacturing costs and thus indicates AM potential (ratio > 20; empirically derived). A ratio close to 1 indicates an almost cubic, geometrically simple component.
- iii. The ratio of the number of component features and the component's surface in cm^2 can be calculated.

This rule describes the complexity of the components through the accumulation of many features in a small area. Components with many features are of interest for AM due to their expensive manufacture with conventional technologies. If this value falls below the empirically derived limit of 0.03 features per cm^2 , the component is not suitable to be additively manufactured.

Further rules of other screening modules.

In addition to the Geometry Module, further fuzzy rules of the three other modules must be applied to quantify the final technology potential of each component. Since the article at hand focuses on the Geometry Module, the holistic evaluation of the overarching part screening methodology and all fuzzy rules will be part of the main author's next publication.

5 Summary and outlook

This work elaborated on a method for an automated geometry analysis of metallic components within a part screening methodology to accelerate product development processes. The resulting *Geometry Module* represents the fourth source of information within the screening approach that compares different manufacturing technologies on a component level. It embodies the following achievements and thus affirms the research questions of Sect. 2.3:

- Picture recognition and classification: 2D image data can be used to classify components by manufacturing technology using a *multi-view* model. The 2D approach ensured low computing times even for a high number of components and images. The classification results led to insights regarding alternative manufacturing technologies that differ from the components' status quo.
- Object recognition within pictures: Holes, bores, and closed profiles can be detected within component pictures and used to support the classification task and the fuzzy rules. The achieved accuracies of over 90% on a picture level are sufficient for the fuzzy approach of the overall methodology. Using the existing images of the classification task ensured a lean data flow within the *Geometry Module*.
- Part screening: The *Geometry Module* was integrated into the data flow of already existing modules of the overarching part screening methodology. The classi-

fication values and component features can be used to expand fuzzy rule sets for the evaluation of manufacturing technology alternatives. The execution of the Geometry Module with over 400 components took 1.5 h. The automatized, image-based approach hence confirmed the intended acceleration of geometric analyses compared to CAD-based methods. The loading and analysis of 3D-parts, e.g., with existing geometric similarity assessment tools in CATIA, took around 1 min per part (on a 16 GB RAM machine), hence 6.5 h for 400 components.

The geometry module's general approach should be transferred and tested in other application fields to ensure its validity. Furthermore, an in depth-comparison of the image-based object detection approach with, e.g. shape segmentation of 3D objects might allow for further insights regarding the trade-off between short computing times and high accuracies. The holistic screening methodology's technology suggestions considering all four modules will be critically examined as part of the main author's next publication.

Appendix: Training and validation of the classification approaches of Sect. 4.1

See Figs. 26, 27, 28, 29 and 30.

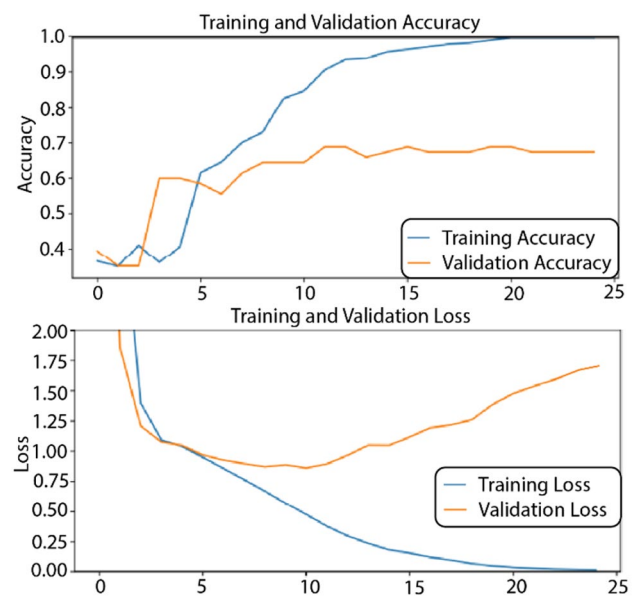


Fig. 26 Training and validation results of the stitching approach, visualization of accuracy and loss

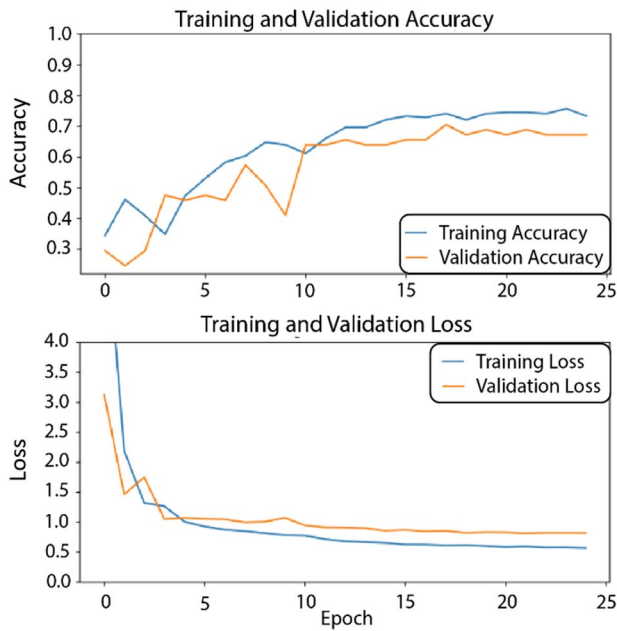


Fig. 27 Training and validation results of the 7NN approach, visualization of accuracy and loss, perspective “bottom”

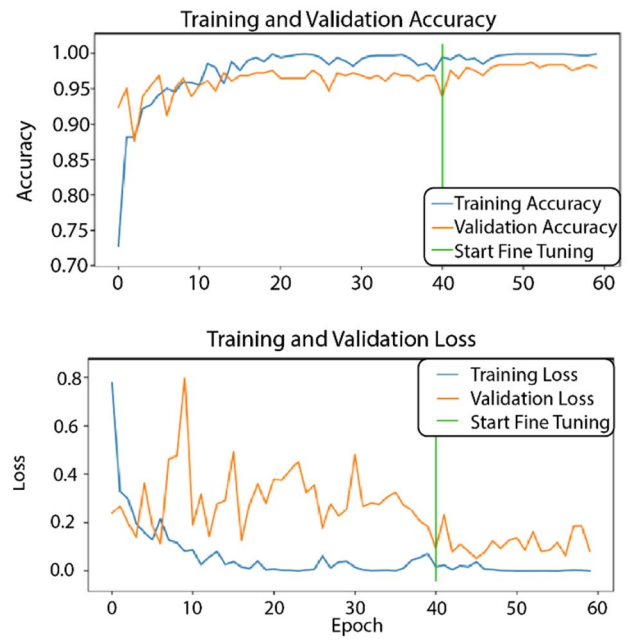


Fig. 29 Training results of the best version of the multi-view approach with data base extension, visualization of accuracy and loss

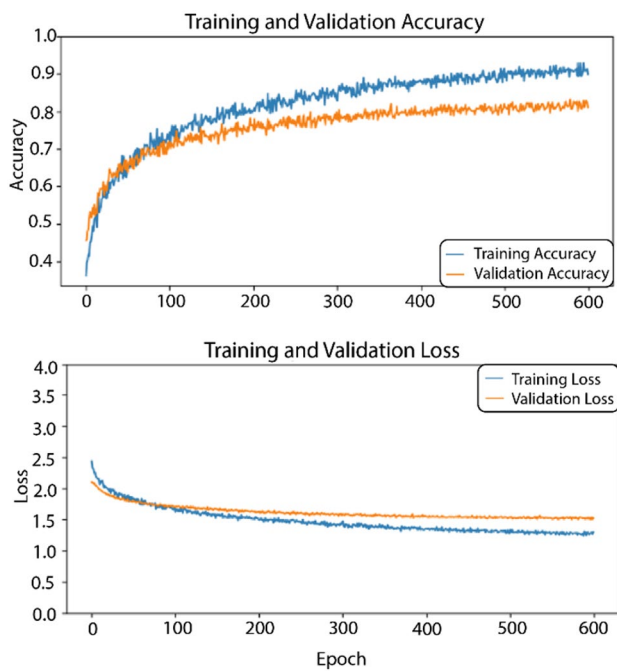


Fig. 28 Training results of version V2 of the single CNN approach, visualization of accuracy and loss

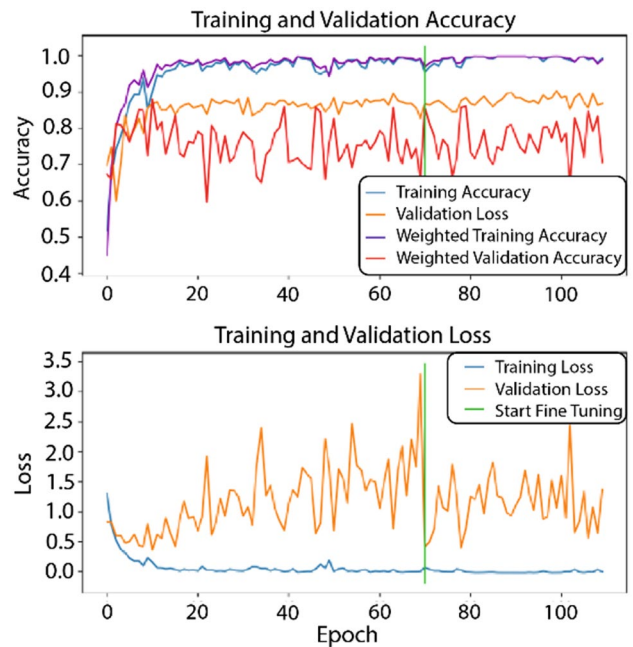


Fig. 30 Training results of the multi-view approach with class extension, visualization of accuracy and loss

Acknowledgements This work was carried by order of the BMW Group in cooperation with the Technical University of Munich.

Funding Open Access funding enabled and organized by Projekt DEAL.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Schmenner RW, Swink ML (1998) On theory in operations management. *J Oper Manag* 17(1):97–113
- Ehrlenspiel K, Kiewert A, Lindemann U, Mörtl M (2014) *Kostengünstig Entwickeln und Konstruieren*. Springer, Berlin. <https://doi.org/10.1007/978-3-642-41959-1>
- Sabadka D, Molnár V, Fedorko G (2019) Shortening of life cycle and complexity impact on the automotive industry. *TEM J* 8(4):1295–1301
- Buechler T, Schumacher F, Reimann P, Zaeh MF (2021) Methodology for an automatic and early manufacturing technology selection on a component level. *Prod Eng Res Devel* 16:23–41. <https://doi.org/10.1007/s11740-021-01070-2>
- Nandhakumar S, Thirumalai R, Viswaaswaran J, Senthil TA, Vishnuvardhan VT (2021) Investigation of production costs in manufacturing environment using innovative tools. *Mater Today Proc* 37(Part 2):1235–1238
- Buechler T, Kolter M, Hallweger L, Zaeh MF (2022) Predictive cost comparison of manufacturing technologies through analyzing generic features in part screening. *CIRP J Manuf Sci Technol*. <https://doi.org/10.1016/j.cirpj.2022.04.012>
- Goodfellow I, Bengio Y, Courville A (2016) *Deep Learning*. MIT Press, Cambridge (ISBN: 978-0262035613)
- Paass G, Hecker D (2020) *Künstliche Intelligenz*. Springer Fachmedien, Wiesbaden. <https://doi.org/10.1007/978-3-658-30211-5>
- Rebala G, Ravi A, Churiwala S (2019) *An introduction to machine learning*. Springer, Cham. <https://doi.org/10.1007/978-3-030-15729-6>
- Hadjileontiadis LJ, Moussavi ZMK (2018) Current techniques for breath sound analysis. In: Priftis K, Hadjileontiadis L, Everard M (eds) *Breath sounds*. Springer, Cham, pp 139–177. https://doi.org/10.1007/978-3-319-71824-8_9
- Janiesch C, Zschech P, Heinrich K (2021) Machine learning and deep learning. *Electron Mark*. <https://doi.org/10.1007/s12525-021-00475-2> (forthcoming)
- Valaskova K, Klietnik T, Misankova M (2014) The role of fuzzy logic in decision making process. In: 2nd international conference on management innovation and business innovation, Bangkok, Thailand. <https://doi.org/10.5729/Inms.vol44.143>
- Kruse R, Borgelt C, Braune C, Klawonn F, Moewes C, Steinbrecher M (2015) *Computational intelligence. Eine methodische Einführung in Künstliche Neuronale Netze, Evolutionäre Algorithmen, Fuzzy-Systeme und Bayes-Netze*. Pp 1–11. ISBN 978-3-658-10904-2
- Wohlkinger W, Aldoma A, Busu RB, Vincze M (2012) 3DNet: large-scale object class recognition from CAD models. *Inst Electr Electron Eng*. <https://doi.org/10.1109/ICRA.2012.6225116>
- Chauhan R, Ghanshala KK, Joshi RC (2018) Convolutional neural network (CNN) for image detection and recognition. In: 2018 first international conference on secure cyber computing and communication (ICSCCC), pp 278–282. 978-1-5386-6373-8
- Fang W, Zhang F, Sheng VS, Ding Y (2018) A method for improving CNN-based image recognition using DCGAN. *CMC* 57(1):167–178. <https://doi.org/10.32604/cmc.2018.02356>
- Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. [arXiv:1409.1556v5](https://arxiv.org/abs/1409.1556v5)
- Hejrati M (2015) *Analyzing 3D objects in 2D images*. Dissertation, University of California, Irvine
- Su H, Maji S, Kalogerakis E, Learned-Miller E (2015) Multi-view convolutional neural networks for 3D shape recognition. In: 2015 IEEE international conference on computer vision (ICCV). IEEE, pp 945–953. <https://doi.org/10.1109/ICCV.2015.114>
- Seeland M, Maeder P (2021) Multi-view classification with convolutional neural networks. *PLoS ONE* 16(1):e0245230. <https://doi.org/10.1371/journal.pone.0245230>
- He K, Zhang X, Ren S, Sun J (2015) Deep residual learning for image recognition. [arXiv:1512.03385v1](https://arxiv.org/abs/1512.03385v1)
- Feng Y, Zhang Z, Zhao X, Ji R, Gao Y (2018) GVCNN: group-view convolutional neural networks for 3d shape recognition. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, Salt Lake City, 18–23 June 2018, pp 264–272. <https://doi.org/10.1109/CVPR.2018.00035>
- Padilla R, Netto SL, Da Silva EAB (2020) A survey on performance metrics for object-detection algorithms. In: *Proceedings of the 2020 international conference on systems, signals and image processing (IWSSIP)*. Published by Paiva AC, IEEE, Piscataway, NJ, pp 237–242. <https://doi.org/10.1109/IWSSIP48289.2020.9145130>
- Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. *CVPR* 2015:3431–3440
- Wang P, Gan Y, Shui P, Yu F, Zhang Y, Chen S, Sun Z (2018) 3D shape segmentation via shape fully convolutional networks. *Comput Graph*. <https://doi.org/10.48550/arXiv.1702.08675>
- Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. *Adv Neural Inf Process Syst* 25(2):2012
- Kalogerakis E, Subhransu M, Melinos A, Chaudhuri, S (2017) 3D Shape segmentation with projective convolutional networks. In: *Proceedings of the IEEE computer vision and pattern recognition (CVPR)*. <https://doi.org/10.48550/arXiv.1612.02808>
- Guan B, Li H, Zhou F, Lin S, Wang R (2021) LGCPNet: local-global combined point-based network for shape segmentation. *Comput Graph* 97:208–216. <https://doi.org/10.1016/j.cag.2021.04.028>
- Li L, Sung M, Dubrovina A, Yi L, Guibas L (2019) Supervised fitting of geometric primitives to 3D point clouds. In: *Conference on Computer Vision and Pattern Recognition (CVPR) 2019*. <https://doi.org/10.1109/CVPR.2019.00276>
- Duanshun L, Feng C (2019) Primitive fitting using deep geometric segmentation. In: *36th international symposium on automation and robotics in construction (ISARC 2019)*. <https://doi.org/10.48550/arXiv.1810.01604>
- Amit Y, Felzenszwalb P, Girshick R (2020) Object detection. In: *Computer vision*. Springer International Publishing, Cham, pp 1–9. <https://doi.org/10.1007/978-3-030-03243-2660-1>
- Redmon J, Divvala S, Girshick R, Farhadi A (2015) You only look once: unified, real-time object detection. [arXiv:1506.02640v5](https://arxiv.org/abs/1506.02640v5)

33. Jana AP, Biswas A, Mohana (2018) YOLO based detection and classification of objects in video records. In: 2018 3rd IEEE international conference on recent trends in electronics, information and communication technology (RTEICT), pp 2448–2452. <https://doi.org/10.1109/RTEICT42901.2018.9012375>.
34. Zhu X, Lyu S, Wang X, Zhao Q (2021) TPH-YOLOv5: improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios. arXiv: 2108.11539
35. Jocher G, et al., (2021) ultralytics/ yolov5: v6.0—YOLOv5n ‘Nano’ models, Roboflow integration, TensorFlow export, OpenCV DNN support. <https://doi.org/10.5281/ZENODO.5563715>
36. Yi C, Kaiheng, Chengmeng, Hao Q, Yiming, Hongliang, Yuan L (2022) Yolov6: the fast and accurate target detection framework is open source
37. Shi P, Qi Q, Qin Y, Scott PJ, Jiang X (2020) A novel learning-based feature recognition method using multiple sectional view representation. *J Intell Manuf* 31(5):1291–1309. <https://doi.org/10.1007/s10845-020-01533-w>
38. Zhang Z, Jaiswal P, Rai R (2018) FeatureNet: machining feature recognition based on 3D convolution neural network. *Comput Aided Des* 101:12–22. <https://doi.org/10.1016/j.cad.2018.03.006>
39. Liu W, et al., (2016) SSD: single shot MultiBox detector. In: Leibe B, Matas J, Sebe N, Welling MBd. (eds) *Computer vision—ECCV 2016*. Published by 9905. Lecture notes in computer science. Springer International Publishing, Cham, pp 21–37. <https://doi.org/10.1007/978-3-319-46448-02>
40. Ning F, Shi Y, Cai M, Xu W, Zhang X (2020) Manufacturing cost estimation based on the machining process and deep-learning method. *J Manuf Syst* 56:11–22. <https://doi.org/10.1016/j.jmsy.2020.04.011>
41. Yoo S, Kang N (2021) Explainable artificial intelligence for manufacturing cost estimation and machining feature visualization. *Expert Syst Appl*. <https://doi.org/10.1016/j.eswa.2021.115430>
42. Sager C, Janiesch C, Zschech P (2021) A survey of image labelling for computer vision applications. *J Bus Anal* 4:91–110. <https://doi.org/10.1080/2573234X.2021.1908861>
43. Fernández A, García S, Galar M, Prati RC, Krawczyk B, Herrera F (2018) *Learning from imbalanced data sets*. Springer International Publishing, New York. <https://doi.org/10.1007/978-3-319-98074-4>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.