Heavy machine parts measurement through deep learning

Sara BALLKOÇI
European University of Tirana, Faculty of Engineering, Informatics and Architecture, Department of Informatics and Technology, Tirana, Albania

Alba ÇOLLAKU
European University of Tirana, Faculty of Engineering, Informatics and Architecture, Department of Informatics and Technology, Tirana, Albania

Ardiana TOPI
European University of Tirana, Faculty of Engineering, Informatics and Architecture, Department of Informatics and Technology, Tirana, Albania

Shahina BEGUM
Malardalen University, Vasteras, Sweden

Shaibal BARUA
Malardalen University, Vasteras, Sweden

Emmanuel WEITEN
Volvo Construction and Equipment, Eskilstuna, Sweden

Abstract

Operational continuity of machinery involves continuously monitoring machinery parts to prevent malfunctions. Recently, it has gained popularity in the heavy industry due to its potential to ensure maintenance and address potential malfunctions before they occur. This project focuses on advancing the "Volvo undercarriage wear inspection and maintenance program." The core of this study is the wear and tear inspection process of the undercarriage parts of Volvo's excavators and it investigates the implementation of deep learning and machine learning techniques, focusing on detecting the undercarriage part of the machine and measuring its deterioration while also aiming to minimize associated costs and labor time. The research starts with a comprehensive collection and preparation of the dataset, ensuring its validity for efficient training while addressing data quality and quantity limitations. A thorough examination and evaluation of the Mask R-CNN model for detecting and segmenting objects is conducted, followed by applying OpenCV for extracting measurements and implementing a template-matching model with a VGG16 network for image classification. The thesis concludes by training and evaluating the Mask R-CNN model three times, showcasing its promising ability to detect and segment the undercarriage part with an accuracy of up to 83.47%. The template matching approach achieved an accuracy of 16.67%, while the OpenCV method demonstrated promising capabilities with an error margin of ±0.5mm. These results indicate that inspection efficiency and accuracy could significantly increase, leading to more timely and cost-effective maintenance decisions. Finally, a validation of the approach is applied and presented in an industrial case study provided by Volvo.

Keywords: *Deep Learning, Computer Vision, Convolutional Neural Network, Mask R-CNN, Instance Segmentation, Object Detection, Image Processing, Data Preprocessing, Data Augma Augmentation, Feature Extraction*

Introduction

In today's fast-evolving industrial world, companies like Volvo Construction promote a more sustainable and environmentally friendly future. Volvo Corporation, a leader in this movement, has long been committed to environmental sustainability and corporate social responsibility. It strives to reduce its ecological footprint while offering innovative solutions that drive growth and prosperity. This commitment to sustainability is the foundation for this thesis proposal with the primary objective of digitalizing the undercarriage parts inspection process. The

existing undercarriage wear inspection is a complex process that faces numerous challenges, including the need for specialized equipment, precision issues, and manual data entry. These challenges increase the risk of errors and consume valuable time and resources. Given that Volvo's main goals are cost-effectiveness and time efficiency, a digitalized inspection process is needed to ensure that machinery parts are regularly inspected and replaced as they deteriorate. Essential for this research is Mask R-CNN, an advanced deep learning architecture that enhances object detection through pixel-level instance segmentation (Potrimba, 2023). By generating a pixel-level mask of the undercarriage part, we can employ various methods to determine whether the UC part has deteriorated and needs replacing. One of the approaches is using OpenCV, a library designed for real-time computer vision and machine learning application (L. Zabulis, 2022), especially for tasks such as machinery inspection in industries, which is well-suited to our problem. Template matching is another technique that identifies small portions of an image corresponding to a template image (Wikipedia, 2023), mainly used for tasks such as image classification. This technique is beneficial for classifying our images as worn out or not. By leveraging these three techniques, a promising digitalized solution can be employed to Volvo, aligning with their cost-effectiveness and time-efficiency goals. This thesis will examine only one undercarriage part, the sprocket (Forestell, 2024). The primary focus of this study is training a Mask R-CNN model on a customized dataset, specifically consisting of images of the sprocket. Based on instance segmentation, Mask R-CNN excels at creating a precise mask for objects, effectively separating the region of interest from the background. Our research indicates that Mask R-CNN outperforms YOLO in terms of accuracy and precision, particularly when detecting each sprocket's teeth on heavy machinery. Subsequently, the generated mask from the model is utilized to assess whether the sprocket is deteriorated. To achieve this, OpenCV is applied to extract measurements from the mask, and a template-matching model is used to classify images as either worn out or not. The proposed approach aims to digitally collect measurements from the sprocket by significantly reducing cost and labor time. The proposed solution has been validated and verified using real-world data from Volvo.

Literature Review

Deep Learning

Deep learning has emerged as a study topic in recent years, driven by the rapidly expanding demand for learnable robots capable of addressing a wide range of complicated problems. The term "deep learning" describes a group of algorithms

built on artificial neural networks specially designed to handle unstructured data, including text, audio, video, and images (Deshpande, 2019). Deep learning (DL) is a subset of artificial intelligence (AI) and machine learning (ML), defined by the usage of neural networks with three or more layers, it enables machines to learn from past experiences autonomously. DL is viewed as an AI function that simulates how the human brain processes information. The term "Deep" in deep learning methodology refers to multiple levels or stages of data processing to create a data-driven model (I.H.Sarker, 2021). As previously stated, deep learning involves the use of multi-layered neural networks. These multi-layered neural networks attempt to imitate the human brain's learning patterns, allowing computers to analyze and learn from massive amounts of data (Awan, n.d.).

Mask Region-based Convolutional Neural Network

Mask R-CNN expands on Faster R-CNN by adding a third output branch that creates masks, thus helping capture more precise spatial features. This improvement adds pixel-to-pixel alignment, a crucial addition missing in Faster R-CNN.

Mask R-CNN architecture

1. Backbone: The model's architecture includes a pre-trained convolutional neural network for feature extraction, commonly a ResNet50 or ResNet101. The first layers detect features of low levels, such as edges and corners, whereas the second and after layers detect features of higher levels (e.g., bicycles, planes, and dogs). When going through the backbone network, the images are converted from 1024x1024px x 3 (RGB) to a feature map of shape 32x32x2048 (K. He, 2017). This feature map will serve as input for the subsequent segmentation phases. The Feature Pyramid Network (FPN) enhances the backbone and represents things at several scales. FPN ameliorates the traditional feature extraction pyramid by incorporating a second pyramid that funnels the high-level features from the first pyramid down to the lower levels. This allows features to be accessed at every level, both higher-level and lower-level features (K. He, 2017).
2. Region Proposal Network (RPN): Region Proposal Network is a compact neural network that examines images using a sliding-window method and finds regions containing objects. The areas the RPN examines are named anchors, which, as shown in Figure 6, are boxes distributed over the image. The RPN rapidly scans many anchors thanks to the sliding window, which is handled by the convolutional nature of the RPN and allows for the simultaneous examination of all regions (typically on a GPU machine). Instead of scanning the image directly, RPN examines over the backbone

feature map, allowing for efficient reuse of the extracted features and avoidance of duplicate calculations. Using these adjustments, the RPN runs at approximately 10 ms (S. Ren, 2016). For each anchor, the RPN creates two outputs:

- Anchor Class: Foreground and background classes comprise the Anchor Class. According to the foreground class, that box probably contains an item (S. Ren, 2016).
- Bounding Box Refinement: The positive foreground anchor may not be precisely centered over the object. To improve the anchor box's fit on the object, the RPN calculates a delta (% change in x, y, width, and height) (S. Ren, 2016).

3. ROI Classifier and Bounding Box Regressor: This stage generates two outputs for each ROI (K. He, 2017):
   - Class: is the class of the object in the ROI. The deep network can classify regions into specific classes (bicycle, dog, balloon,...etc.). It also generates a background class, which discards the ROI. • Bounding Box Refinement: Later on, this is used to fine-tune the bounding box's position and dimensions to enclose the item entirely. ROI Pooling Variable input sizes are difficult for classifiers to handle. Usually, they demand a set amount of input. However, the ROI boxes may differ in size due to the bounding box refinement phase in the RPN. ROI pooling is useful in this situation. Like cropping and resizing a portion of an image, ROI pooling involves part of a feature map being cropped and then resized to a defined size. The ROI Align approach, proposed by the authors of Mask R-CNN (K. He, 2017), involves sampling the feature map at different places and using a bilinear interpolation.

OpenCV

Library OpenCV (open-source computer Vision collection) is a collection of programming functions designed primarily for real-time computer vision and machine learning applications. The library is cross-platform and available as free and open-source software under the Apache Licence 2. Since 2011, OpenCV has supported GPU acceleration for real-time operations (OpenCV, 2024). OpenCV was initially intended to provide a standardized infrastructure for computer vision activities. Still, it has since played an essential role in advancing the integration of machine perception into commercial goods. Notably, the Apache 2 license promotes accessibility and adaptability, making it easier for enterprises to use and modify. OpenCV has approximately 2500 optimized algorithms and covers many traditional and cutting-edge computer vision and machine learning techniques. These algorithms provide various functions, including face

detection and recognition, object identification, human activity classification in movies, camera movement tracking, and 3D model extraction (K. He, 2017). Furthermore, OpenCV is important in image stitching, 3D point cloud generation, and augmented reality application development. Its large user base exceeds 47 thousand people, and an estimated download count of over 18 million demonstrates its widespread use in various industries. Major organizations such as Google, Microsoft, Intel, and countless startups rely heavily on OpenCV for their projects. The library's applications range from street view picture stitching and surveillance video intrusion detection to robot navigation and product inspection in factories worldwide. OpenCV has interfaces for C++, Python, Java, and MATLAB, and it supports various operating systems, including Windows, Linux, Android, and macOS (OpenCV, 2024).

Template Matching

Template matching is a method used in digital image processing to identify small portions of an image that correspond to a template image. Applications of this technology include edge detection in photos, mobile robot navigation, and industrial quality control (Wikipedia, 2023). The feature-based template matching method uses deep neural networks such as Convolutional Neural Networks (CNNs) like VGG, AlexNet, and ResNet to extract visual properties, including shapes, textures, and colors. These networks generate features for matching at hidden layers by producing vectors with picture categorization information. Robust and efficient, this approach can handle light changes, background clutter, and non-rigid transformations (Wikipedia, 2023). In contrast, the template-based method works well when the templates fully depict the matched image or don't have any distinguishing characteristics. To produce multi-scale representations from preprocessed images, it is necessary to reduce the search space to achieve effective matching. The curse of dimensionality in machine learning datasets can be alleviated, and the number of sampling points can be reduced with the help of this technique (Wikipedia, 2023).

Related Work

Mask R-CNN with custom dataset

Instance segmentation is challenging as it involves accurately recognizing all objects in an image and precisely segmenting each instance. Several instances of segmentation-based deep learning techniques have been proposed recently. The YOLOV3 object detection technique, which combined candidate bounding box

prediction and feature extraction into a single deep convolutional network, was first presented by Redmon et al. (J. Redmon and A. Farhadi, 2018). He et al. (K. He, 2017) introduce an extension of the Faster R-CNN framework by adding a segmentation mask prediction branch to improve the accuracy of object instance segmentation. This method efficiently accomplishes precise segmentation at the pixel level, beating previous approaches on benchmarks such as COCO, by using a tiny, fully convolutional network to each Region of Interest (ROI) (K. He, 2017). Integrating Generative Adversarial Networks (GANs) with instance segmentation models represents a unique technique for improving instance segmentation accuracy, as presented by Le et al. (Q. H. Le, 2021). This study substantially alters the Mask R-CNN framework by integrating a discriminator network that assesses the quality of segmentation masks created by the segmentation network, which is conceived as a generator. They move from pixel-based to feature-based processing by using the segmentation network as a generator and adding a discriminator that assesses mask quality. This modification leads to improved segmentation performance and more stable training dynamics, especially when handling complicated images with various object forms and a lot of clutter (Q. H. Le, 2021). Their approach proves the usefulness of adversarial loss in doing away with the requirement for domain-specific tuning by exhibiting notable improvements in defining crisp boundaries and detailed object features across various applications, including autonomous driving, cellphone recycling, and medical image analysis. Zhang et al. (C. Zhang, 2022) proposed another instance segmentation model for steel defect detection based on the traditional Mask R-CNN model. To improve the accuracy of the Mask R-CNN model, the authors replaced the feature extraction network with a more efficient and robust EfficientNet for extracting features of different scales. Also, a CBAM module was added to the mask branch to enhance the quality of mask prediction. Several experiments were conducted on the Severstel steel defect dataset, resulting in improved accuracy and efficiency (C. Zhang, 2022). Sun et al. (G. Sun, 2022) proposed a new technique for cropping and extracting information from images based on an improved Mask R-CNN model by incorporating the Softer NMS algorithm. This paper creates the backbone network for feature extraction and target candidate region generation by combining ResNet50 and FPN. This method removes unnecessary anchor frames, optimizes the feature pyramid network's (FNP) structure, and increases crop detection accuracy (G. Sun, 2022).

Template matching for detecting defects

Yuan et al. (W. Yuan, 2023) propose a defect detection method for detecting defects on photolithography masks, which present a significant challenge regarding their size from 1 to 2 pixels. The proposed approach combines template matching in

spatial and frequency domains for accurate registration, minimizing error from registration deviations (W. Yuan, 2023). The experiment also includes a variational model and a primal-dual optimization algorithm to compute the point spread function. The last step of the approach is comparing the gray values between the sample and the modified template images, using threshold extraction to identify the defect areas, a technique proven effective through testing (W. Yuan, 2023). For detecting defects on personalized printings, a new method called secondary template matching was introduced by Ma et al. (B. Ma, 2017) as the complex and irregular surface of the printings presents challenges while using conventional template matching algorithms. The method includes aligning the image under inspection with a template image to identify ROI (region of interest). After that, a four-threshold algorithm separates the image into foreground and background. Finally, the secondary template matching algorithm is applied separately to identify defects, resulting in effective defect detection on personalized printings (B. Ma, 2017).

Object Measurements Using Computer Vision

OpenCV (Open Source Computer Vision Library) is an essential tool for computer vision, extensively used for object measurement across various industries. An example of the application of OpenCV is automating inspections in modern manufacturing, aiming for Industry 4.0. The authors Zabulis et al. (L. Zabulis, 2022) discuss using a CV system to measure the length of objects as they move along a conveyor. The method includes image segmentation using a convolutional neural network, custom contour analysis, and the least square method to determine the midline of wooden planks. The experiment was evaluated using a dataset of wooden planks from an actual EURO pallet. With a length measurement accuracy of 1mm and a processing time of 0.97 seconds, the experiment confirmed to function as anticipated (L. Zabulis, 2022). Another important aspect of computer vision in today's industries is the real-time detection and measurement of objects. The authors Othman et al. (N. A. OTHMAN, 2018) introduce a refined method for object detection and measurement in real-time video streams. Their proposition includes using OpenCV libraries that leverage canny edge detection and dilation and erosion processes. The experiment consists of four essential steps: detecting the objects using canny edge detection, applying dilation and erosion, identifying and organizing contours, and measuring the dimensions of the objects (N. A. OTHMAN, 2018). The tools used to deploy this experiment are a Rasberry Camera, a Rasberry Pi3, and an OpenCV library. The results of this novel technique had a successful rate of approximately 98% in measuring the objects' size (N. A. OTHMAN, 2018).

Methodology

Literature Review

A systematic literature review will be conducted using databases such as Google Scholar, IEEE Explore, and ACM Digital Library, alongside source materials provided by Volvo. The thesis will use critical terminology related to Volvo's heavy machinery, Machine Learning, and fault detection using image processing, Convolutional Neural Networks, and Computer Vision. The study aims to identify existing methodologies, algorithms, and techniques for object detection and degradation or deterioration prediction.

Data Collection and Preprocessing

Volvo Construction will give us access to its databases for data collection, providing us with the necessary data for the research. The database consists of diverse images from various machine lifecycles and wear stages. After data collection, the images will be annotated and prepared for the training phase.

Mask R-CNN Model Implementation and Training

Development phase: design and construct the Mask R-CNN model as the suitable algorithm for object detection and instance segmentation due to its dual capabilities in object detection and pixel-level instance segmentation, which are critical for accurately assessing the condition of machinery parts. Mask R-CNN's adaptability allows us to customize the model to meet the unique needs of our dataset, which eventually results in a more precise and dependable answer (K. He, 2017).

Training phase: A transfer learning technique will refine the pre-trained model on the Coco dataset, significantly reducing training time and enhancing the model's ability to generalize data unique to Volvo. This approach is a vital part of our methodology, ensuring the model is well-equipped to handle the specific challenges of our research.

Testing on Unseen Data

Evaluate the trained model's performance on unseen images.

Measurements extraction using OpenCV

After generating masks from the trained model, OpenCV will retrieve measurements. This is an essential step of the research, requiring accurate measurements beneath the created masks. Integrating these two techniques, OpenCV and Mask R-CNN is ideal for creating a reliable and effective inspection system.

Template matching implementation using VGG16

We will use the VGG16 pre-trained network and a template-matching approach to identify worn-out photos. his classification step will facilitate accurate wear condition identification

Results Analysis

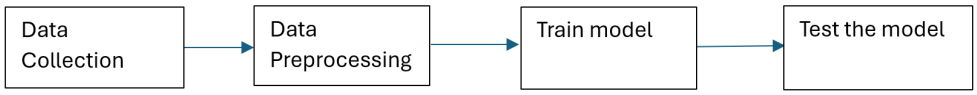
The results obtained from the experiments will be analyzed to evaluate the proposed solution's effectiveness and accuracy. The analysis will examine performance metrics such as confusion matrix, average precision, and mean average precision for the Mask R-CNN model, the precision of measurements obtained using OpenCV, and the classification accuracy of the template matching approach with VGG16.

Discussion and Conclusion

The data and observations gathered during the research project will be thoroughly analyzed and discussed. he proposed software solution's advantages, potential drawbacks, and future work will be discussed, and comparisons with current solutions will be made

Methods and Analysis

The method used is divided into two sections: the first involves training and evaluating the Mask R-CNN model, a cutting-edge approach to segmentation. The second section focuses on extracting measurements from images to calculate the wear stage and predict future degradation of the machinery. The first section involves the steps below:



Data Collection

The first step we took while implementing our thesis was data collection. As a crucial collaborator in this research, Volvo Corporation has played a significant role by providing us with the dataset that forms the foundation of our implementation. The dataset provided by Volvo consists of 150 images and 30 videos capturing various excavator models at different life cycle stages. Particular emphasis is placed on images showing machines in various levels of wear and tear. This includes information gathered over time via their inspection procedures. For the development of this thesis, we are focused on model EC210B, which offers four distinct machine operating hours (4894, 2750, 3969, and 3902 operating hours). As part of our comprehensive approach, we collected images from machines in new condition, with 0 operating hours, conveniently available in Volvo's office. We tried to take as many pictures as possible from different angles and distances. However, it soon became clear that the data provided was insufficient. In response to limited data availability, approximately 4000 images were extracted from provided videos using the OpenCV library, providing a more extensive image dataset.

Data Preprocessing

Our first step in developing the thesis was annotating the data/ images we had secured from Volvo. Image annotation is a crucial component of our dataset preparation procedure while using the Mask R-CNN algorithm because it provides ground truth data that the model uses during training to learn to detect and segment objects accurately and support our primary goal of segmentation.

There are multiple tools that you can use for data annotations, such as: LabelMe, RectLabel, LabelBox, VGG Image Annotator (VIA COCO UI). We used the LabelME tool to annotate the images. Through this thorough process, we annotate and label the sprocket teeth to provide the necessary training data for our model. These annotations will later be used to obtain measurements and calculate the wear stage of the UC part, a crucial step in our research. Selecting the appropriate amount of samples for the training, validation, and testing datasets was a crucial phase in developing our thesis. Numerous considerations influenced this choice, including the size of Volvo's dataset and the requirement to guarantee an adequate amount of data for thorough model testing and training. Initially, the dataset comprised images and videos of various excavator models at different life cycle stages. However, because there was a shortage of data, we used the OpenCV package to extract about 4000 images from the given videos to increase the dataset. This resulted in a much larger and more diverse dataset, guaranteeing a more thorough depiction of the excavator parts' wear and tear situations. Subsequently, the annotated dataset was partitioned as follows:

- *Training Set:* 290 images were used to train the Mask R-CNN model. This set is the primary source of information that the model uses to identify patterns and characteristics connected to the sprocket teeth.
- *Validation Set:* For validation, we employed 70 annotated photos. This subset helps avoid overfitting by enabling the model's hyperparameters to be adjusted and offering an intermediate performance check.
- *Testing Set:* Comprised of 70 unseen images for the model, allocated to evaluate the trained model.

The dataset was split into an 80% training set and a 20% validation set, with the testing set being the same size as the validation set. Utilizing transfer learning eliminates the need for an extensive dataset to train our deep learning model. This approach leverages a pre-trained weights file already trained on the Microsoft Everyday object in context (MS COCO dataset), which includes around 80 object classes. Consequently, the pre-trained weights have already learned many features, making it easier for the model to recognize the teeth of the sprocket. Following the preprocessing steps, this dataset provides the fundamental framework for the initial stage of creating and assessing the optimal Mask R-CNN model utilizing the dataset.

We trained our model starting from the COCO weights file. The COCO dataset, designed to advance image recognition, contains approximately 80 common object categories, with 82 of them having more than 5,000 labeled instances. It has 2,500,000 labeled instances in 328,000 images and fewer categories but more for each category. By leveraging this dataset, we enhanced the accuracy and efficiency of our model, a crucial step in our research process.

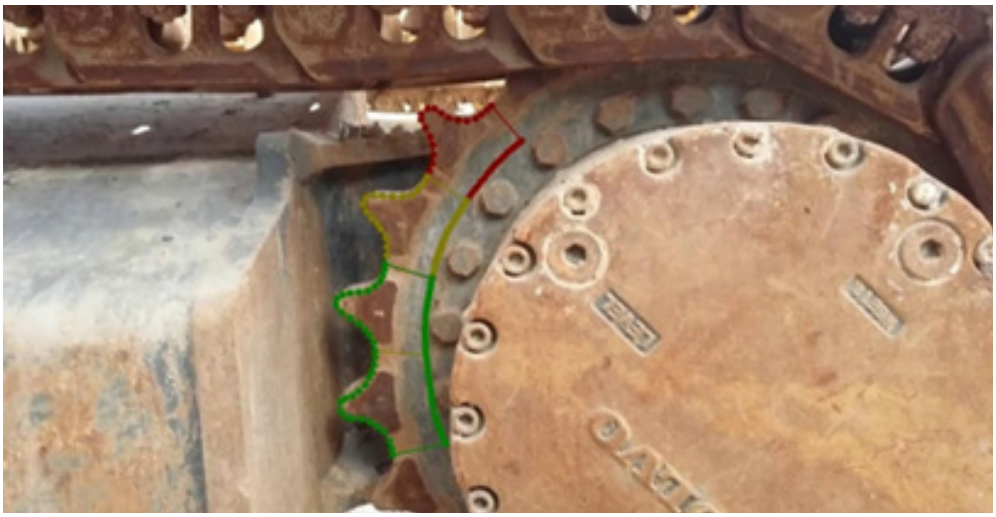
Mask R-CNN implementation

We implemented Mask R-CNN using Python 3.8.18, Keras, and TensorFlow 2.4.0, based on the Feature Pyramid Network (FPN) and ResNet101 backbone. This setup is essential for capturing high-resolution details at multiple scales, particularly in contrast to our photos' intricate and diverse backgrounds. It is also required for accurately recognizing and dividing the excavator's teeth' various shapes and sizes. Mask R-CNN includes a pre-trained CNN for feature extraction, usually using ResNet50 or ResNet101. Our implementation of Mask R-CNN uses ResNet101 over ResNet50 because it offers a more profound layer structure, and it can learn more complicated characteristics and enhance detection accuracy in complex images. This is useful for processing the different-sized and shaped teeth of the excavator against highly textured and textured backgrounds. Our dataset includes annotated images of the excavator's teeth taken from multiple angles and in varied lighting circumstances during training. This ensures the model's robustness and reliability in real-world operational settings.

The model is fine-tuned using this dataset, a process that adapts the pre-trained MS COCO dataset to the characteristics of our task, such as detailed segmentation and precise localization of each tooth for practical wear analysis and maintenance prediction. The ROI Align layer enhances the model's ability to resolve alignment problems in previous models by carefully extracting features from every suggested region. Mask R-CNN's dual-head system allows each tooth to be detected simultaneously and precisely segmented. One head is in charge of bounding box regression and object classification, and the other is responsible for mask prediction. This segmentation capability is crucial for applications requiring precise delineation, such as automated wear analysis or predictive maintenance duties, since it provides comprehensive outlines of each tooth through binary masks.

By leveraging the pre-trained weights from the MS COCO dataset, we sped up the initial training process, reducing the need for a large volume of training images and computational resources. This method conserved a great deal of processing power and time. It gave our model a wealth of prelearned features from the outset, improving its capacity to generalize from the small amount of task-specific training data. Pre-trained models have several advantages over training from scratch, especially regarding training efficiency and initial accuracy. They have learned features usually applicable over an extensive range of pictures, thereby the network can quickly adjust to the unique features of new datasets with little extra input.

FIGURE 1: The process of annotating the images



Training phase

The training phase for the Mask R-CNN model was conducted using a laptop equipped with an Intel Core i5-6300U CPU @ 2.40GHz 2.50 GHz supported by 8GB of RAM. The training was with 360 images split into 290 images for train 70 for val, and it was run for approximately 86 hours (3 days and 14 hours).

Template matching using VGG16

In this approach, we are focused on assessing whether an object, in this case, a tooth, is in good condition or worn by comparing the similarity of two images given as input. We chose template matching because it is widely implemented in many research papers for comparing parts of a source image against a template image, which may be an image of an object, and it serves as a template to detect similar objects in the source image. It compares items to industry standards and identifies flaws. For extracting features from the photos, we used VGG16. This pre-trained convolutional neural network has been initially trained on the ImageNet dataset, a large visual dataset that includes around 20000 object classes and more than 14 million annotated images. It utilizes a technique called deep template matching, using a pre-trained VGG16 model. This model can classify images based on their similarity to a reference image. Firstly, a reference image is loaded and preprocessed using preprocessed API in Keras for VGG16. Features from the reference image are extracted using the VGG16 model. Then, the new image,

which needs to be checked for quality, is loaded. The image features are extracted using the same VGG16 model. Table 4 shows the default parameters of VGG16 network. We used the same default parameters for our implementation as they applied to our solution. The only parameter that we changed was the threshold.

Extracting Measurements from OpenCV

As part of achieving our goal of providing a rough estimate of the wear of an excavator's teeth, after the model recognizes the teeth, it runs a script that uses OpenCV to measure the height of a tooth. The two main libraries of the script are OpenCV, a library dedicated to computer vision tasks, and numpy, which is used for numerical operations and, in our case, for distance calculations. To accurately measure the dimensions of the tooth from an image, we used a pixel-to-millimeter conversion method based on the information in the blueprint provided to us by VOLVO. First, we identified the height of the tooth (22 mm) as our reference measurement. Using an interactive script, we measured this height in the image, 160.86 pixels. By dividing the pixel measurement by the known physical height, we calculated the Pixels Per Millimeter (PPM) ratio, which was approximately 7.31 pixels/mm. With the PPM established, we could measure any distance in the image and convert it to millimeters. Then, we wrote another script that allowed us to view the distance in pixels between these points and convert this pixel distance to millimeters using the PPM ratio. The script also annotated the image with pixel and millimeter measurements, ensuring clear visualization and accurate dimensional analysis. This method provides a reliable means of translating pixel measurements into real-world dimensions for precise analysis.

Results

Instance Segmentation

The training period was used to fine-tune the model's capacity to recognize and categorize the "tooth" component of excavator undercarriages using picture data. The session was carefully planned to address the faults of the preceding one, with particular objectives established to raise the model's accuracy, precision, and recall. We examined the confusion matrix, precision-recall curves, and mAP in various circumstances and contexts to make specific conclusions about the model's efficacy and dependability in practical situations. AP at IoU=50 is a metric that evaluates the precision of the model when the predicted bounding boxes overlap the ground truth boxes by at least 50%. Meanwhile, mAP is used to evaluate the performance of object detection models across multiple classes and IoU thresholds, and a higher mAP value indicates better performance. A lower mAP indicates lower model performance. These assessments provide insight into

the model's capabilities and serve as a roadmap for upcoming changes to enhance its functionality. The following subsections thoroughly explain each training step and show how the model's performance changed over time as it was refined.

- Confusion matrix

The confusion matrix analysis shows that the model's predictions were easily understood. The model obtained 101 True Positives (TP) for the "tooth" class, which means it correctly detected 101 instances of teeth. There were also 20 False Negatives (FN), in which the model failed to detect teeth that were present, and 24 False Positives (FP), in which the model mistakenly classified nontoothed regions as teeth. This resulted in a precision of 80.80% for the "tooth" class, demonstrating that the model is highly accurate when predicting a tooth. The recall for the "tooth" class was 83.47 %, indicating that the model successfully detected 83.47% of all actual teeth in the dataset.

These metrics underscore the model's capability to detect teeth accurately and highlight areas for improvement in reducing false positives and negatives.

- Precision recall curve

The precision-recall curves provided further insights into the model's performance under different scenarios. One precision-recall curve exhibited a perfect AP@50 of 1.0, indicating that the model maintained high precision and recall across various thresholds for a specific subset of the data. This demonstrates the model's potential to achieve almost perfect detection under optimal conditions. However, another precision-recall curve showed a significantly lower AP@50 of 0.115, highlighting the variability in the model's performance. This suggests that while the model performs exceptionally well in some cases, it struggles in others, possibly due to variations in image quality, lighting, or other challenging conditions.

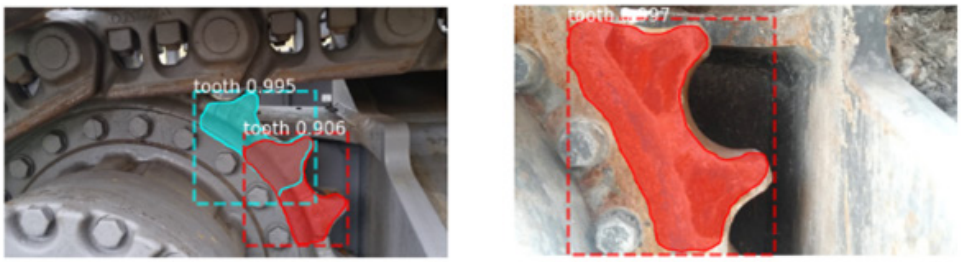
The detailed mAP values for individual images often revealed consistently high performance. For example, several images achieved an AP of 1.0, showcasing the model's ability to detect and localize teeth accurately in these cases. This high level of performance was maintained across numerous images, contributing to the overall high mAP score. However, there were also instances with lower AP values, indicating that the model's performance can fluctuate based on specific image characteristics. The evaluation results demonstrate that the Mask R-CNN model effectively detects the "tooth" class in a custom dataset.

The model's high mAP of 0.9076, combined with a precision of 80.80% and recall of 83.47% for the "tooth" class, illustrates its strong capability in accurately identifying teeth. The precision-recall curves further confirm the model's

robustness, with one curve showing a perfect AP@50 and another indicating areas for potential improvement. These results collectively highlight the model's strengths while also pointing to opportunities for further refinement to enhance its performance consistency across diverse and challenging scenarios. To conclude, the Mask R-CNN model has demonstrated good precision and recall, showing promising results in detecting the "tooth" class. The model is highly effective under some settings. Still, as seen by the variety in precision-recall curves, it will require continuous improvements to guarantee consistent performance across all test photos.

The below figure shows the detection result we got after testing unseen images after the third training. The masked part displays the "teeth" of the sprocket detected by the Mask R-CNN model. The containers and labels mark the masked area in which we are interested in this implementation. The bounding box labeled "tooth" is followed by a value that indicates the model's confidence in the objects' accurate detection. The dashed red line represents the range where the detection was concentrated.

FIGURE 2: The results of the model after training



Template Matching using VGG16

When the developed template matching using a VGG16-based image classification model was tested with a dataset of 30 source images, comparing them to a template image in good condition, only 5 of these 30 images were correctly classified as worn. This result corresponds to an accuracy rate of approximately 16.67%. The result shows a notable difference in the model's prediction accuracy, and it highlights the need for additional modifications or investigations to the model's architecture, feature extraction, and classification methods to increase the system's dependability in precisely recognizing worn states. This result implies that the feature extraction technique or threshold setting may not be optimally adjusted for the wear characteristics of the employed photos. The dataset employed is another significant factor in this outcome since Volvo's images lacked quality and uniformity in terms of pixel quality.

Results of extracting measures using OpenCV

The methodology enabled precise measurement of the excavator tooth's dimensions directly from the image. After identifying the tooth's height in the picture as 160.86 pixels, we calculated the Pixels Per Millimeter (PPM) ratio of approximately 7.31 pixels/mm. We could accurately convert pixel measurements to millimeters by applying this PPM ratio. The accuracy of this method was validated through repeated measurements and multiple modifications to the code, demonstrating its reliability for precise dimensional analysis in engineering applications. Extracting the measurements from an image using OpenCV involves several vital steps to ensure accuracy and precision. First, after loading the image using the first script, we select specific points to create the reference we use in the main script. The points would then be used to measure the distance using the Euclidean distance formula, where x1 x2 y1 y2 in the formula represents the coordinates of the two selected points. This formula ensures accurate calculations of the straight-line distance between the points in the image. A known reference measurement from the blueprint determines the PPM ratio to convert the pixel measurements into real-world dimensions. This is calculated using the formula:

PPM = Pixel Height / Physical Height

For instance, if the measured pixel height of the tooth is 160.86 pixels and the known physical height is 22 mm, the PPM is calculated as follows:

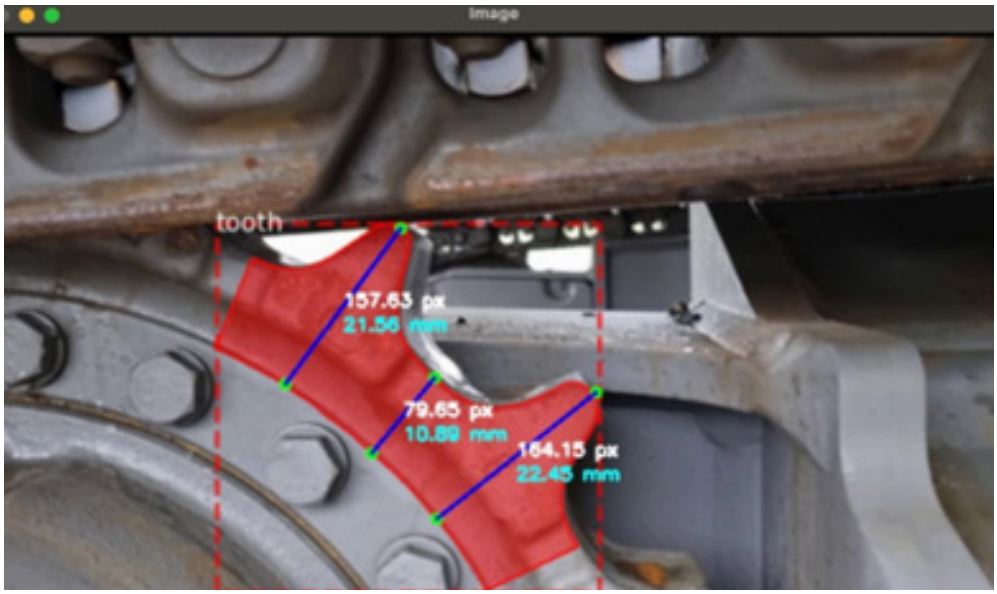
PPM = 160px / 22mm = 7.31 Pixels per mm

This PPM ratio is then applied to convert any measured pixel distance to millimeters using the formula:

Distance (mm) = Distance(Pixels) / PPM

The accuracy of these measurements is validated through repeated trials and adjustments to the code, ensuring reliable dimensional analysis. Despite the high precision, potential sources of error include image resolution pixel interpolation. To validate that the script is accurate, we tested it on multiple images of excavator teeth with little to no wear. After numerous trials with it, we have concluded that we have only a margin of about ±0.5 mm, which is a reasonable estimate that shows the script's accuracy. The figure below illustrates how this program works.

FIGURE 3: The results of the OpenCV method



Bibliography

1.Awan, A. A. (n.d.). *What is Deep Learning? A Tutorial for Beginners*. Retrieved from https://www.datacamp.com/tutorial/tutorial-deep-learning-tutorial

2.Deshpande, V. K. (2019). Deep learning, in Data Science (Second Edition), Chapter 10.

3. Forestell, K. (2024, January). Exploring the Anatomy of an Excavator: A Guide to Its Essential Parts. Retrieved from DOZR: https://dozr.com/blog/parts-of-an-excavator.

4. He, K., Gkioxari, G., Dollar,P., & Girshick, R.(2017). Mask R-CNN. Computer Vision and Pattern Recognition Journal, (pp. 2961–2969). https://doi.org/10.48550/arXiv.1703.06870.

5. Le, H.Q., Youcef-Toumi, K., Tsetserukou, D., Jahanian, A. (2021). Instance semantic segmentation benefits from generative adversarial networks. Computer Vision and Pattern Recognition Journal. https://doi.org/10.48550/arXiv.2010.13757.

6.Ma, B., Zhu, W., and Wang, Y.(2017). *The defect detection of personalized print based on template matching. 2017 IEEE International Conference on Unmanned Systems (ICUS). Proccedings book* (pp. 266–271).

7. Othman, A.N., Salur,U.M., Karakose, M., Aydin, I. (2018). An embedded real-time object detection and measurement of its size. International Conference on Artificial Intelligence and Data Processing (IDAP) 2018, Turkey.

8. OpenCV. (2024). Retrieved from Wikipedia: https://en.wikipedia.org/w/index.php?title=OpenCV&oldid=1208982530

9. Potrimba, P. (2023). What is Mask R-CNN? The Ultimate Guide. Retrieved from roboflow: https://blog.roboflow.com/mask-rcnn/.

10. Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement.
DOI: 10.48550/arXiv.1804.02767.
11. Ren, Sh., He, K., Girshick, R., Sun, J. (2016). Faster R-CNN: Towards real-time object detection with Region Proposal Networks. Computer Vision and Pattern Recognition Journal.
https://doi.org/10.48550/arXiv.1506.01497.
12. Sarker, H.I. (2021). Deep learning: A comprehensive overview on techniques, taxonomy, applications and research directions.SN Computer Science Journal, Volume 2, article number 420, (2021).
13. Sun, G., Wang, Sh., Dong, L., Du, Y.(2022). Crop image segmentation method based on improved Mask RCNN. MSIE 2022: 2022 4th International Conference on Management Science and Industrial Engineering.
14. Yuan, W., Zhou, Y., Luo, C. (2023). Defect detection in masks based on variation and template matching. 2023 IEEE 6th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC). Proccedings book pp. 48–54.
15. Zabulis, L., Lipnickas, A., Augustauskas, R. (2022). Application of computer vision methods for
automated wooden planks length measurement. 18th Biennial Baltic Electronics Conference (BEC), pp 1-6.
16. Zhang, C., Yu, B., Wang,W. (2022). *Steel surface defect detection based on improved MASK RCNN*.
2022 IEEE 8th International Conference on Computer and Communications (ICCC).
17.Wikipedia. (2023). *Template matching*. Retrieved from Wikipedia:
https://en.wikipedia.org/w/index.php?title=Template_matching&oldid=1135654295