Automating Log Sweep Assessment:

A log sweep detection model based on video imagery and deep learning

AUTHOR:
Perry Han

SUPERVISORS:
Professor Rien Visser
Professor Glen Murphy

Executive Summary

This study aims to support the development of an automated system for measuring log curvature ('sweep') using computer vision. Detection of log sweep has relied on manual measurement which is very inefficient, or the experience of machine operators. To date there is no accurate and fast way to determine the sweep of logs, leading to potential misclassification and reduced value recovery at time of harvest.

This project shows how computer vision and deep learning can build a model that automatically identifies and calculates log sweep. The model was built based on the deep learning framework of Pytorch and the deep learning environment of MMDetection.

Data was collected and pre-processed to train the model. A total of 100 images were manually annotated using LabelMe software. The annotated JSON files were programmatically transformed into coco-type files for deep learning. The pre-processed data was divided into a training set, a validation set, and a test set in a ratio of 4:1:1. This ratio avoids overfitting of training results.

The image parameters were modified based on the existing Mask R-CNN base model and training was started to obtain a model that only detects the stems in the image. The coordinates of all pixels of the mask can be extracted from the detection results. Using the coordinates that make up the shape of the log, a simple mathematical model can then calculate sweep.

Most of images evaluated were more than 90% accurate in identifying the stem, which was the desired goal. The programme currently calculates a relative sweep, which is the ratio of the offset of the centre-line to straight line as drawn from the two end mid-points. In future work, accuracy can be improved by, for example, adjusting the brightness or contrast of the image and enhancing the existing model. By interfacing with the harvester head that captures log length and small end diameter, the sweep can be referenced against the log sweep specifications to see if it adheres to the expected standard. The future goal is to achieve real-time, fully automated video detection and sweep calculation for harvesting and processing operations.

TABLE OF CONTENTS

1. Introduction

The New Zealand forest industry has moved away from manual to mechanised felling as technology and equipment have developed for harvesting timber. The processing of timber at the harvesting site is no exception and approximately 95% of the on-site processing of timber is now carried out by mechanised processor heads (Figure 1).

There are many types of processor heads on the market today, and most processor heads can measure length and diameter accurately. Specifically, it uses three parameters to support the log-making decisions: small end diameter, large end diameter, and quality codes entered manually by the operator. These heads also have software that automatically optimises log making decisions by predicting the expected log recovery options.

Figure 1: Waratah processor head

Even though processor heads have been automated in many features, the measurement of log's 'curvature' is not currently automated. The curvature of logs is also known as the 'sweep' in the forest industry, and sweep can currently only be measured manually, or is dependent on experienced operators when processing timber.

There is an excellent prospect for automating the sweep of logs, and the realisation of this goal can significantly increase the value recovery and productivity of the forest industry.

2. Literature Review

Achieving automation of sweep measurement requires many knowledge backgrounds to support it. In this section, the necessary knowledge will be explained. Some of the content will not be explored in depth as specific literature can be read to understand the construction of some software or data models. The focus will be on how to use these necessary techniques to achieve the goal.

2.1. Log Sweep Definition

There are two methods of measuring the sweep of logs, and they explain how the sweep is calculated. In general, the sweep is the maximum deviation from straightness along the length of the log. Method 1 (Figure 2) is to measure the distance between the log and the tape at its widest point. In practice, hold the tape taut and across the inside of the bend and then measure the distance in centimetres between the two points. If the tape is held inside the bark at each end, the gap should be measured inside the bark.

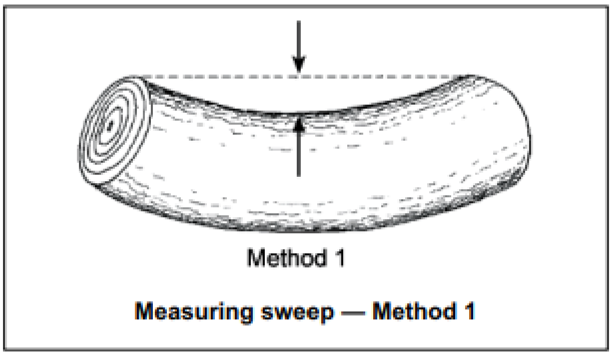

Figure 2: Sweep Measurement Method 1 (Competenz, 2017)

Method 2 (Figure 3) is to hold the tape firmly along the trunk between the centres of the two ends of the log. Then, measure the maximum distance between the tape and the centre line. There is typically only a minor difference between the two methods for logs with a small amount of tape. However, the difference can be significant for logs with large taper.

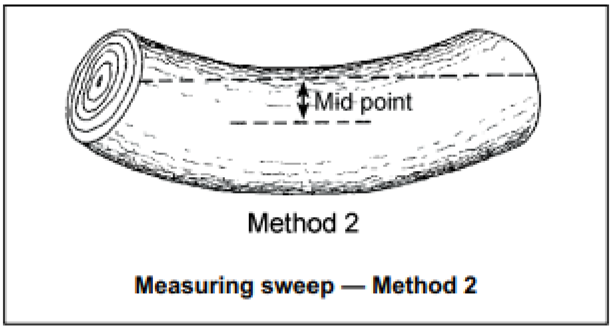

Figure 3: Sweep Measurement Method 2 (Competenz, 2017)

While manual measurement of the sweep of logs is still an effective method of measuring sweep, an example of the potential problem is that novice operators will be asked to manually check the accuracy of the first few logs' sweep grade. Each day, they need to go through this process when they start processing logs, as shown in Figure 4 below.

Figure 4: Manually Measuring Log Sweep (Hamner, 2007)

The sweep of logs has not only its calculation method but also a strict grading of specifications. As this project aims to improve logs' value recovery and productivity, Tables 1 and 2 indicate different New Zealand domestic grades requirements.

Table 1: New Zealand Domestic Grades (Maclaren, 2000)

1a: Domestic grades

Log grade	Pruned?	Minimum small-end diameter mm	Maximum knot size mm	Sweep class*
P1	Pruned	400	0	1
P2	Pruned	300	0	1
S1	Not necessary	400	60	1
S2	Not necessary	300	60	1
S3	Not necessary	200	60	1
L1	Not necessary	400	140	1
L2	Not necessary	300	140	1
L3	Not necessary	200	140	1
Pulp	Not necessary	100	No limit	1 or 2

Table 1 shows that the majority of domestic log sweep specifications are classified as Class 1. This grade is also more stringent compared to Class 2. Table 2 also shows that sweep is restricted according to the length of the log, with different lengths having different maximum allowances.

Table 2: New Zealand Sweep Classification for Domestic Grades (Maclaren, 2000)

1b: Maximum permissible sweep*

Sweep class	Log length			
	<3.7 m	**3.7–4.8 m**	**4.9–7.6 m**	**>7.6 m**
Class 1	d/8	d/4	d/3	d/2
Class 2	d	2d	3d	4d

d = small-end diameter of log.
*Sweep is the maximum deviation from straightness along the length of the log.

There is another influence to be considered for this project. No actual log lengths will be measured for this project, and no automatic length measurement will be used. Therefore, the sweep ratio as a new concept needs to be introduced to classify the log sweep. The sweep ratio is the ratio of the deflection to the small end diameter of the log. The comparative data at the pixel level for both the deflection and the small end diameter are automatically calculated in this project. The ratio can therefore be compared with the multiplication constants of the small end diameters in Table 2. It allows the specification to be graded in terms of the actual log lengths on the joint. In terms of the formula,

$$Deflection \leq parameter \times SED$$

It will transfer to function below,

$$Sweep\ \ Ratio = \frac{Deflection}{SED} \leq parameter$$

The sweep ratio will also be used as the final result of this project as it will be easier to interface with the software on the processor's head for automatically grading the specifications later on.

There is not only a domestic New Zealand sweep specification grading for different logs. There is also a similar grading for export, but more stringent (Table 3). One noteworthy point is lifting the restriction on the sweep in exporting pulp logs, which is a comparative difference to domestic specifications.

Table 3: New Zealand Export Grades (Maclaren. 2020)

2: Export grades

Log grade	Small-end diameter mm	Maximum large-end diameter mm	Maximum knot size mm	Length m	Percentage allowed*	Sweep
Pruned peelers	300+	No limit	0	4.0 6.0	Shipper's option	d/4
Japan A	200–340	800	d/3 up to 150 mm maximum. Excessive number of large knots not permitted	4.0 8.0 12.0	10% balance 50% minimum	d/4 d/2 d
Japan J	200–260	No limit	As above	4.0 8.0 12.0	As above	As above
Korea K	200–260	No limit	As above	3.6 5.4 7.3 11.0	10% maximum balance 40% minimum	As above
Pulp	100+	No limit	No limit	4.0 6.0 8.0	Shipper's option	No limit

d = small-end diameter.

2.2. Computer Vision and Artificial Intelligence

Computer vision and artificial intelligence will be used to develop an automatic log sweep recognition system. Therefore, a lot of knowledge related to computer science and software engineering will be used. Computer vision is the study of how to make machines "see". It refers to cameras and computers to identify, track and measure targets instead of the human eye. It is primarily used to simulate the superior capabilities of human vision and compensate for human vision deficiencies.

The superior ability to simulate human vision can be seen in many ways. Human vision can recognise various object scenes and people. They estimate three-dimensional space and distances to enable navigation by avoiding obstacles, understanding and interpreting pictures. Bridging the gaps in human vision means improving the problem that humans can focus on the salient content and thus miss many details. It can also be thought of as improving human vision that describes subjectivity, ambiguity, inability to perform the same task over time consistently.

Artificial intelligence is a way of making computers, computer-controlled robots or software think intelligently. It's similar to the way that intelligent humans think. AI is the study of how the human brain thinks and how humans learn, make decisions and work when solving problems. Then, using the results of that study as the basis for developing intelligent systems.

Computer vision is closely related to artificial intelligence, but it is also fundamentally different. Artificial intelligence places greater emphasis on reasoning and decision making. However, computer vision is still mainly at the stage of image information representation and object recognition. Object recognition and scene understanding also involve reasoning and decision-making from image features but is fundamentally different from AI reasoning and decision-making.

The relationship between the knowledge frameworks used in the project can be represented in the diagram below as shown in red. Each layer is explained step by step in the following sections.

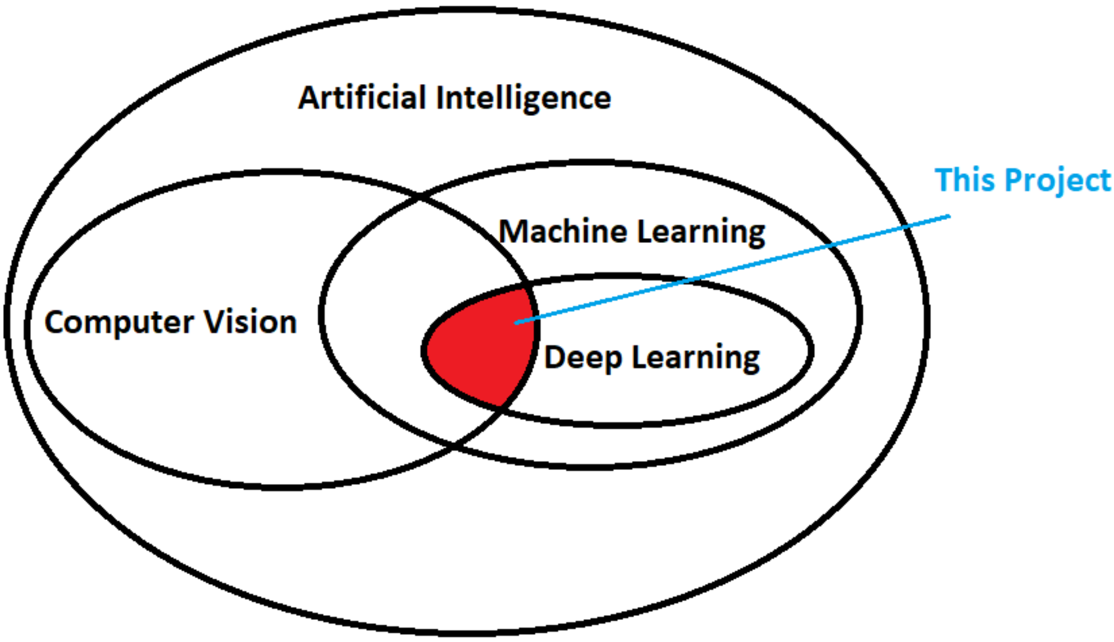

Figure 5: Knowledge Relationships

2.3. Machine Learning

Machine learning is a method to achieve artificial intelligence, similar to data mining. It is a multi-disciplinary subject involving probability theory, statistics, approximation theory, convex analysis, computational complexity theory and many other disciplines. In contrast to data mining, which looks for mutual properties between big data, machine learning focuses more on the design of algorithms that allow computers to learn patterns from data and use them to make predictions about unknown data. Because learning algorithms involve a great deal of statistical theory and are particularly closely linked to statistical inference, they are also known as statistical learning methods.

Machine learning studies how computers can simulate human learning behaviour to acquire new knowledge or skills and reorganise existing knowledge structures to improve themselves continually. Computers learn patterns and patterns from data to apply to new data for predictive tasks. Algorithms for machine learning can be divided into three main categories: supervised algorithms, unsupervised algorithms, and reinforcement algorithms (Thomas W. Edgar, 2017).

Supervised Algorithms: A model can be learned or built from the training dataset during supervised learning training, and new instances can be inferred from this model. The algorithm requires a specific input/output and first decides what kind of data to use as an example. It is like a handwritten character in a text recognition application or a line of handwritten text. The main algorithms include Neural Networks, Support Vector Machine, K-Nearest Neighbors, Naive Bayes Model, and Decision Tree.

Unsupervised Algorithms: These algorithms do not have a specific target output; the algorithm divides the data set into different groups.

Reinforcement Algorithms: Reinforcement learning is universal and is mainly based on decision making. The algorithm trains itself based on the success or error of the output and will give better predictions after a lot of practical training. In this way, an organism is stimulated by rewards or punishments given by the environment, gradually forming stimulus expectations and generating habitual behaviour. The most significant benefit will be obtained. In operations research and cybernetics, reinforcement learning is known as approximate dynamic programming (ADP).

The main thing used in this project is a neural network system to classify supervised learning algorithms. A neural network is a computational model with connected node layers and a hierarchical structure similar to a network of neurons in the brain. Neural networks can learn from data and be trained to recognise patterns, classify data, and predict future events. This concept will be explained in more detail in the following Deep Learning section.

2.4. Deep Learning

Deep learning is an important branch and extension of machine learning and is a neural network structure containing multiple hidden layers. Deep learning learns the most critical features of the data itself by combining lower-level features to form more abstract higher-level representations of attribute categories or features. Simply put, deep learning is a deeper, more complex neural network structure.

As with machine learning methods, there are supervised and unsupervised learning approaches to deep machine learning. The learning models built under different learning frameworks are very different. For example, Convolutional Neural Networks (CNNs) is a machine learning model under deep supervised learning. In contrast, Deep Belief Nets (DBNs) are a machine learning model under unsupervised learning. This project will focus on CNN deep learning model, which is unsupervised.

Deep learning enables complex function approximation by learning a deep non-linear network structure that characterises a distributed representation of the input data and demonstrates a powerful ability to learn the essential features of a data set from a small sample set. The benefit of multiple layers is that complex functions can be represented with fewer parameters.

By gradually transforming the initial low-level feature representation into a high-level feature representation through multi-layer processing, complex learning tasks such as classification can be accomplished using simple models. Deep learning can thus be understood as feature learning or representation learning.

The essence of deep learning is to attain more valuable features by building machine learning models with hidden layers and large amounts of training data that ultimately improves the accuracy of classification or prediction. Thus, deep models are the methods, and feature learning is the object. Deep learning emphasises the depth of the model structure. It highlights the importance of feature learning by transforming the feature representation of a sample in the original space to a new feature space, layer by layer, thus making classification or prediction easier. The use of big data to learn features can portray rich inherent information than manual rule-based methods of constructing features (Emmert-Streib Frank, 2020).

2.4.1. Convolutional Neural Network (CNN)

Convolutional Neural Network is one of the most representative network structures in deep learning technology. It has been very successful in image processing, with many successful models based on CNN on the international standard ImageNet dataset. One of the advantages of CNNs over traditional image processing algorithms is that the complex pre-processing of the image (extraction of artificial features, etc.) is avoided, and the original image can be fed directly. A Convolutional Neural Network is a feed-forward neural network containing artificial neurones that respond to a subset of the surrounding units in the coverage area. It is ideal for large-scale image processing.

Convolutional Neural Networks comprise many convolutional layers, a top fully connected layer with relative weights and a pooling layer. It is comparable to a classical neural network. Convolutional neural networks can use this structure to utilise the two-dimensional system of the input data. Convolutional neural networks outperform other deep learning architectures in image and voice recognition. A back-propagation approach may also be used to train this model. Convolutional neural networks require fewer parameters to be evaluated than other deep, feed-forward neural networks, giving them an appealing deep learning structure (Stanford University, 2020).

2.4.2. Recognition and Segmentation

This project aims to automate the calculation of the sweep of a target log, so the shape of the target log needs to be automatically detected and identified. It is where the concept of instance segmentation comes into play. In deep learning image analysis, the four steps are shown in Figure 6 below clearly explain the relationship between the various image analysis methods.

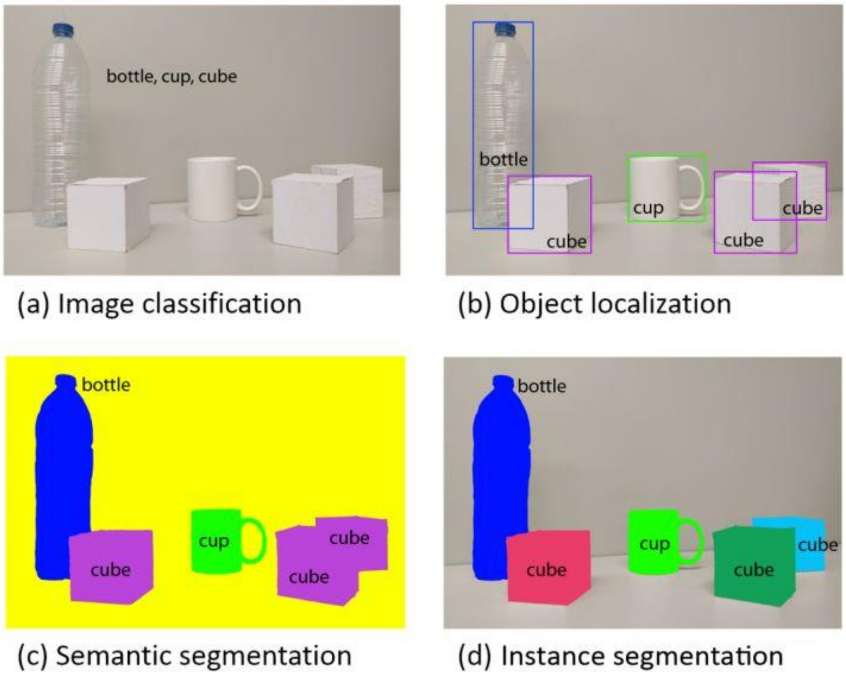

Figure 6: Recognising Objects. (He K. G., 2017)

a) Image classification: It indicates what is inside the target image
b) Object localisation: This step will first detect the objects in the image and combine them with the classification to recognise the detected object.
c) Semantic segmentation: The image will be segmented by serval objects. It can show the shape of detected objects.
d) Instance segmentation: Segmentation by individual objects based on individual targets.

Semantic segmentation assigns a category to each pixel in an image, but objects between the same categories are not distinguished. Instance segmentation, on the other hand, only classifies specific objects. It looks similar to target detection, with the difference that target detection outputs the bounding box and category of the target, and instance segmentation outputs the mask and category of the target (Fengting Yang, 2020).

2.4.3. PyTorch

There are now two main deep learning frameworks, PyTorch and Tensorflow. This project is based on PyTorch because it has various advantages; being easy to use because it has less abstraction, more intuitive design, a transparent and straightforward modelling process and easy to understand code. More details on deep learning implementations such as back-propagation and other training processes are available to users. Also, this framework has a lively community and there is documentation and guidelines available. The author maintains a forum for users to communicate and ask questions. The code is very Pythonic (clean and elegant) and has better debugging features. The default run mode is more like traditional programming, and standard debugging tools such as PDB, IPDB, VSC or PyCharm debugger can be used. The process of training neural networks is straightforward when using PyTorch, and it supports data parallelism, distributed learning models and includes many pre-trained models. It is better suited to small projects and prototyping. PyTorch's dominance in research continues to grow. Figure 7 above shows the ratio of papers using PyTorch to those using Tensorflow or PyTorch at the top research conferences. Overall usage is on the rise and rising very fast, with documents implemented in PyTorch for every major conference in 2019.

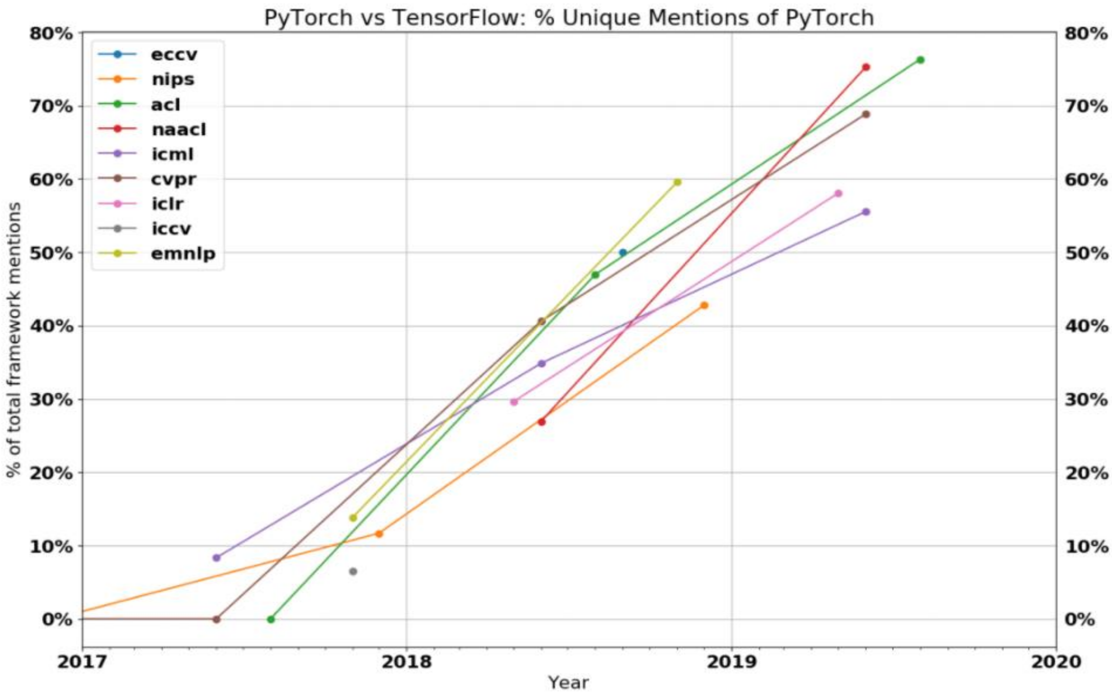

Figure 7: PyTorch vs TensorFlow: % Unique Mentions of PyTorch (He H. , 2019)

2.5. Mask R-CNN

Mask R-CNN is the preferred basic model which could process all steps above. R-CNN or Region-CNN is the first algorithm to successfully apply deep learning to target detection, based on algorithms such as convolutional neural networks (CNN), linear regression, and support vector machines (SVM). The Mask R-CNN follows the idea of the Faster R-CNN and uses the ResNet-FPN architecture for feature extraction, with an additional Mask prediction branch. The Mask RCNN is a synthesis of many of the best previous research results. This concept can be written into a formula which will be,

$$\text{Mask RCNN} = \text{Faster RCNN} + \text{ResNet} + \text{FPN} + \text{Mask}$$

2.5.1. Faster R-CNN

The Faster RCNN is a two-stage target detection algorithm consisting of a region proposal in stage 1 and a bounding box regression and classification in stage 2. Figure 8 below visualises the basic structure of the Faster RCNN.

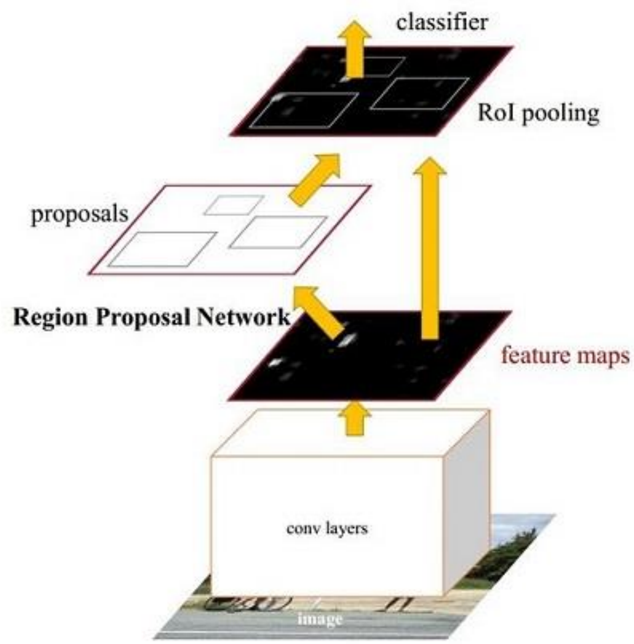

Figure 8: Faster R-CNN Basic Structure (Shaoqing Ren, 2016)

Faster RCNN uses CNN to extract image features, then uses region proposal network to extract ROIs, then uses ROI pooling to turn all these ROIs into fixed size, and then feeds them to the fully connected layer for bounding box regression and classification prediction. There is only a brief overview of this model given. However, the Faster RCNN itself is very detailed and more complex than the first stage algorithm and cannot be described in a few words. The focus here is on a brief understanding of the basic ideas behind the prediction of this model.

2.5.2. Residual Network (ResNet)

As the network deepened, there was a drop in the accuracy of the training set. It is not due to overfitting, as the training set should be highly accurate in the case of overfitting. Traditional convolutional or fully connected networks have loss of information when passing information. There is also the problem of causing gradient disappearance or explosion, which prevents the network from converging and results in intense networks that cannot be trained (Jiayun Wang, 2020).

The Residual Neural Network was proposed by four Chinese, including Kaiming He from Microsoft Research, which successfully trained a 152-layer neural network by using the ResNet Unit. It also won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2015 competition, with an error rate of 3.57% on top5. The number of parameters is lower than

VGGNet, and the results are very outstanding comparison. The comparison graph can be seen in Appendix A (He K. Z., Deep residual learning for image recognition., 2016).

The structure of ResNet can accelerate the training of the neural network extremely fast, and the accuracy of the model has been improved relatively significantly. The main idea of ResNet is to add directly connected channels to the network, which is the idea of the Highway Network. The idea of ResNet is also very similar to the Highway Network in that it allows the original input information to be passed directly to the last layers, as shown in the figure below.

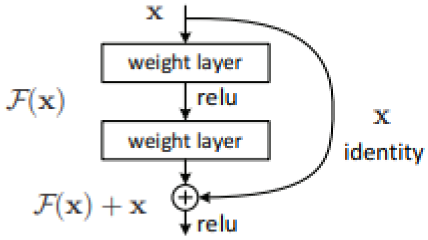

Figure 9: A Building Block of Residual Network (He K. Z., 2016).

In this way, instead of learning the entire output, this layer of the neural network learns the residuals of the previous network's output, hence the name ResNet. The residual network is characterised by its ease of optimisation and its ability to improve accuracy by adding considerable depth. Its internal residual blocks use jump connections to mitigate the gradient disappearance problem associated with increasing depth in deep neural networks (He K. Z., Identity mappings in deep residual networks, 2016).

2.5.3. Feature Pyramid Network (FPN)

Feature Pyramid Network is a top-down feature fusion method, and it is a multi-scale target detection algorithm, i.e., there is not only one feature prediction layer. Although some algorithms also use multi-scale feature fusion for target detection, they use only one scale of features obtained after fusion. This approach can combine the semantic information of the top-level features with the detailed knowledge of the bottom-level features. Still, it will cause some bias in the process of feature deconvolution, etc. (Tsung-Yi Lin, 2017). Using only the features obtained after fusion for prediction will adversely affect the detection accuracy. FPN methods start from the problems mentioned above and can predict multiple fused components at different scales to maximise detection accuracy.

2.5.4. Mask

The Mask R-CNN is built by adding a layer to each RoI of interest for predicting the separation mask, known as the mask branch. The specific framework structure is given in Figure 10 below, parallel to the existing box and classification layers.

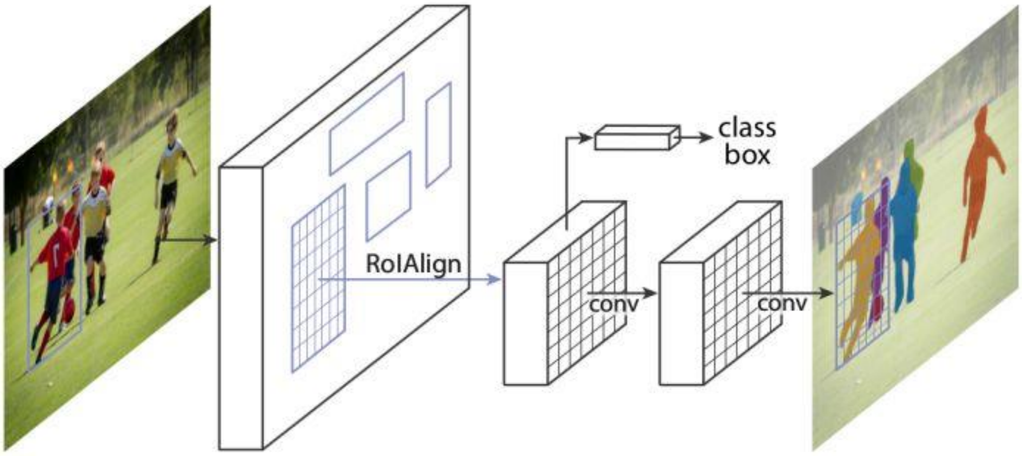

Figure 10: Mask R-CNN Segmentation (He K. G., 2017)

The idea of Mask R-CNN is simple. Since the Faster R-CNN object detection works very well, each candidate region can output kind labels and localisation information. By adding another branch to the Faster R-CNN and adding an output, namely the object mask. The classification and regression tasks transfer to classification, regression, and segmentation. Mask R-CNN combines the binary cover with the classification and bounding box from Faster R-CNN to produce a surprisingly accurate separation of images. As shown below, Mask R-CNN is a flexible and versatile framework for segmenting object instances, which detects targets in an image and outputs a high-quality segmentation result for each target. In addition, Mask R-CNN can be easily generalised to other tasks, such as character keypoint detection.

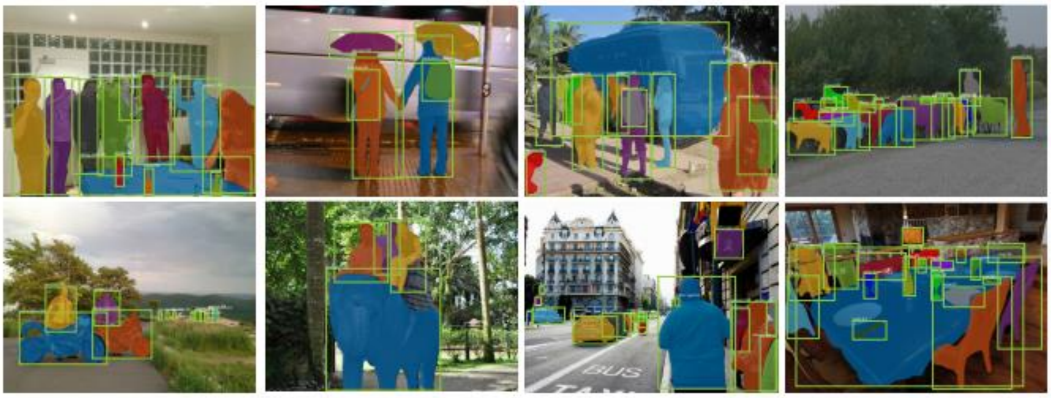

Figure 11: Mask R-CNN Result (He K. G., 2017)

Mask R-CNN can achieve instance segregation at the pixel level and better instance segregation by processing object detection and target segregation simultaneously and in parallel. It splits classification prediction and masks prediction into two network branches, which is similar to Faster R-CNN. The Faster R-CNN gives predictions for regions of interest, produces category labels and rectangular box coordinates. The mask prediction branch creates a binary mask for each category that relies on the classification prediction results, based on which objects are separated. Mask R-CNN independently predicts a binary show for each class, avoiding competition between classes, separated pixel by pixel. There are lots of other networks that can be considered, so the reason why choosing Mask R-CNN are:

- Improved RoI confluence so that the alignment of candidate regions and convolutional features does not lose information due to quantisation by bilinear differences.
- In segmentation, Mask R-CNN decouples the two tasks of judging categories and output templates (mask). It uses a sigmoid with a logistic loss function to process each category template separately. It achieves better results than the classical segmentation method using SoftMax to make all categories compete together (Zhaojin Huang, 2019).

2.6. MMDetection

The MMDetection tool is used here the MMDetection tool is used here to make it easier to implement these complex programming projects. MMDetection is a PyTorch-based target detection toolkit developed by Open-mmlab, a very famous toolkit in target detection like Detectron2. Open-mmlab is a lab jointly established by the Chinese University of Hong Kong, Shang Tang, and other institutions. MMDetection currently contains dozens of models and methods and provides a suitable high-level interface for developers to use as the basis for the secondary development of projects. There have been many papers in the field of detection implemented using MMDetection in the past few years.

3. Methodology

3.1. Image Angle Selection

The project's idea is to use computer vision to figure out the stem grabbed by the processing head, then analyse the specific part of the log to obtain the sweep ratio. Two cameras are needed for this implementation to calculate the correct sweep ratio. This is because a log is a three-dimensional object and one camera can only perform a two-dimensional analysis. Because of practical considerations, the actual two cameras will be placed above and to the side of the target logs. The ideal camera setting position is shown in Figure 12 below.

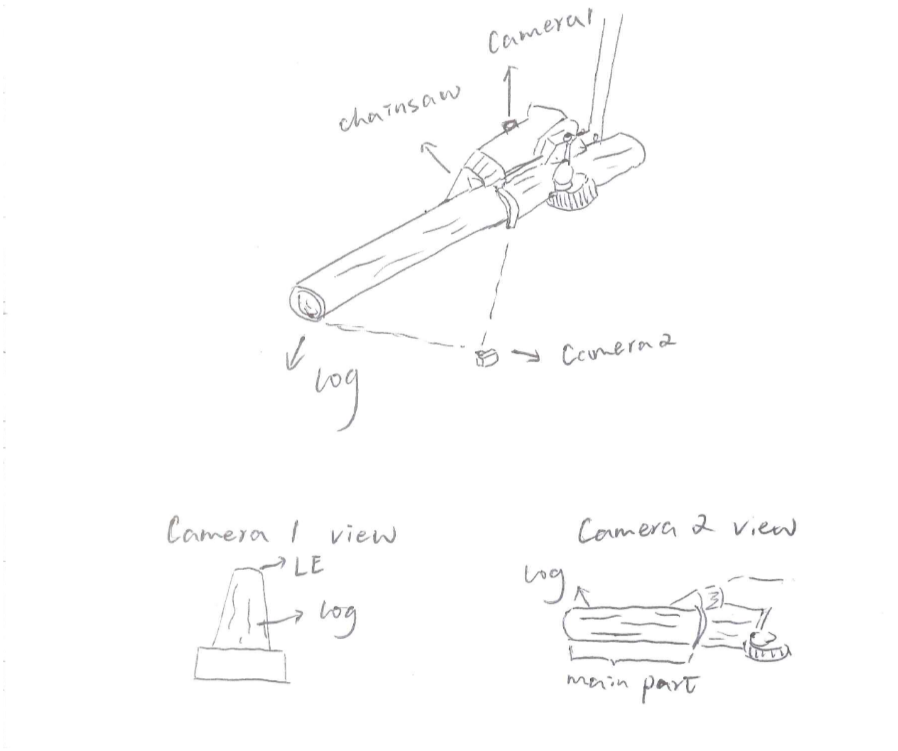

Figure 12: Ideal Camera Setting

Camera one has been designed to be placed above the head of the processor to monitor the logs in a top-down view. Camera two is positioned on the side of the log. It can be fixed to the outside of the processor's cab to give an overview of the whole log, allowing simultaneous inspection of the log with the operator. Therefore, the images used in this project were collected from two directions.

3.2. Data Collection

The method used to obtain the data was to capture picture from online videos. It provided most data resources, but the quality of data is relatively low. An alternative method would have been collecting the data at logging sites resulting in high-resolution pictures and videos that are easier for data analyse, but takes more time and cost. Hence all image or video data was taken from publicly available YouTube sources. The data covers a variety of log species from around the world, and the different types of processors' heads used. Only a small amount of data was from local New Zealand sources and of lower quality. A total of 100 intercepted images were used for model training in this project.

3.3. Coding Environment Construction

Many programming tools were used in this project, and the packages' versions are listed here.

- System Platform: Linux (Ubuntu 20.04)
- GPU: One NVIDIA GeForce GTX 1080 Ti (11GB)
- CUDA: 11.1
- GCC: 7.3
- Python: 3.7.11
- PyTorch: 1.8.0
- Torch Vision: 0.9.0
- OpenCV: 4.5.3
- MMCV: 1.3.9
- MMDetection: 2.15.0+90e9ec7

3.4. Manual Labelling

On the collected image data, this project was manually annotated using LabelMe software. The target logs in each image were labelled individually. The labelling of one of the images is shown in figure 13 as an example.

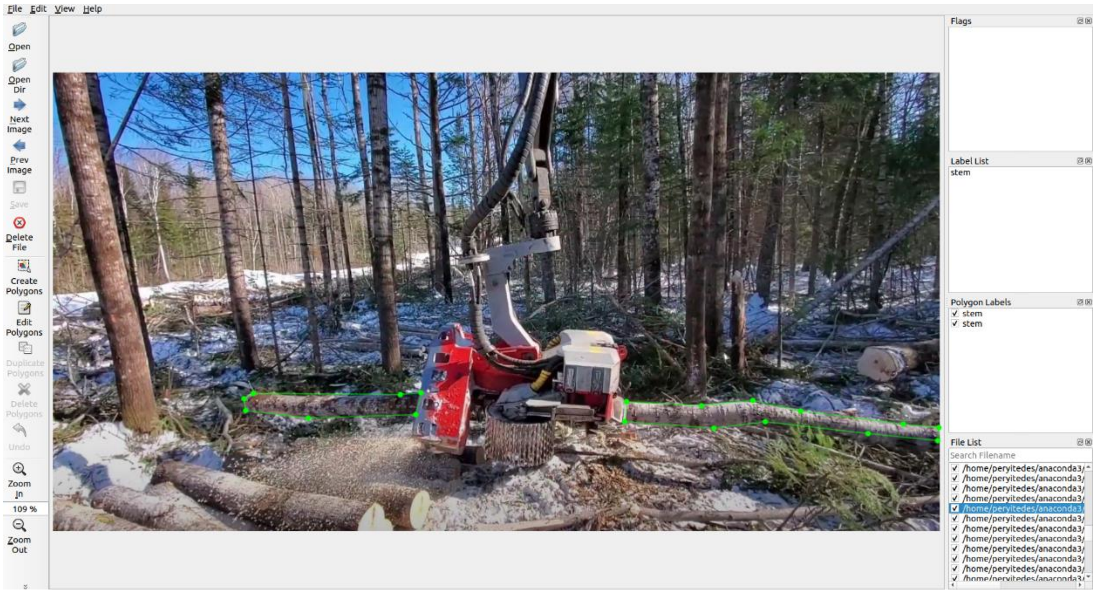

Figure 13: LabelMe Manual Labelling

3.5. Dataset Construction

In this project, the training set, validation set, and test set are separated from the pre-processed data in a ratio of 4:1:1. The pre-processed data were manually annotated using LabelMe and then converted from JSON to COCO format files for use. A total of 100 images from different angles were used for labelling and training.

The training set is the data set used to train the parameters within the model. The validation set is used to tune the hyperparameters. It determines which hyperparameters have the best performance based on several validation sets and monitors the model for overfitting. The test set is used to evaluate the generalisation ability of the model. It means that the previous model uses the validation set to determine the hyperparameters, and a new dataset that has never been seen before is used to determine if the model works.

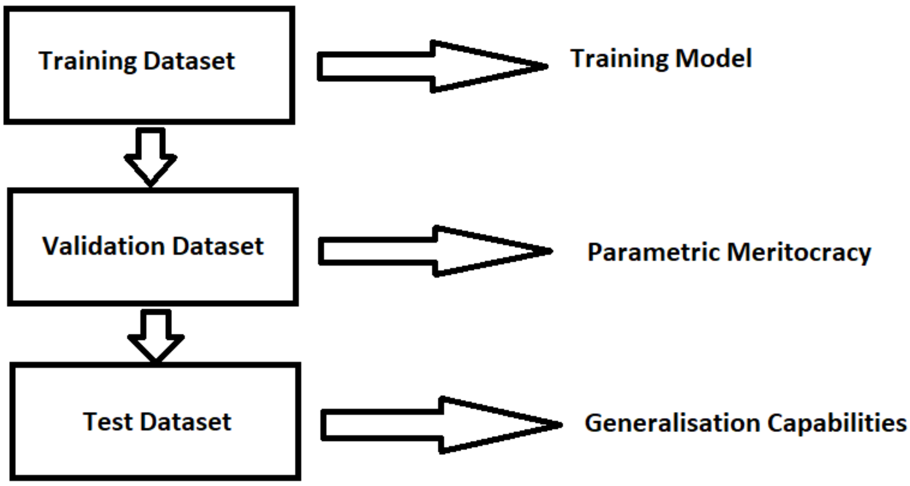

Figure 14: Three Type of Datasets

The training set is like a student's textbook, where the student learns from the textbook. The validation set is like a practice assignment, where the student learns and progresses faster or slower. Then, the final test set is like an exam, where the questions are generally not seen, and the student's ability to learn by example is examined. The training set is directly involved in tuning the model, which obviously cannot be used to reflect the actual ability of the model so that some students who have memorised the textbook will have the best results, which is overfitting and not valid. Similarly, because the validation set is involved in the manual tuning process, it cannot be used as a final judgment of a model. It is as if a student who brushes up on a question bank is not good. So, the test set is used to examine the actual ability of a student's model.

In general, there is a risk of overfitting the test set by dividing the training and test sets only. The sample dataset consists of a training set, a validation set and a test set, where the training and validation sets are for parameter selection of the learning model and the test set is for testing the generalisation ability of the model.

3.6. Training Model

This project used computer vision and supervised deep learning to achieve its goals. The deep learning framework for training the models will be PyTorch, not Tensorflow, for the reasons explained above. The project has also decided, after some testing, to use MMDetection as the overall coding framework, which supports several model templates, including Mask R-CNN. The training profile for this project is based on a modification of the Mask R-CNN model. After the construction of the dataset is completed, the model parameters are refined to achieve the desired goal.

3.7. Result Analyses

Test the new unprocessed data on the completed training model. The data results for the mask and the target detection frame are obtained and visualised in the image. This data result is then extracted and analysed to obtain the coordinates of all pixel points of the mask in the original data map.

Conversely, a fitted line of these coordinates and a straight line at the midpoint of the two ends can be found. The most significant difference between these two lines is the pixel-level deflection. There is a marked position which is 5% of the total x-axis length from each end. The slightest difference in the y-axis between two sides marked positions of all the pixel points is the pixel-level small end diameter. If the inspection is carried out from a top-down angle, the x and y-axis parameter settings must be swapped. Finally, the sweep ratio of the detected logs is obtained.

4. Result

The results of the project were divided into two main phases. The first is to make a pre-trained log recognition model. The second is the extraction and analysis of data based on the trained recognition model, using pixel point coordinates as the primary tool for calculating the sweep ratio of the detected logs. The five sets of results presented in this section are all from the test set.

4.1. Pre-trained Log Recognised Model

This trained model is specifically designed to detect the stems captured by the processor's head. The results are generated as an image that contains the log. Figures 15 to 19 show the orange frame is the rectangle in which the target logs are located, and the text and numbers in the top left corner of the rectangle are the category and accuracy of the target. The accuracy rate means how sure the model is that the object is the log. The light blue mask is an example of the segmentation of the target. In many cases, it can achieve an accuracy of over 90%. However, due to the size of the dataset used (100 manually annotated images), the generalisability and accuracy of the model can still be improved.

Figure 15: Recognising Model Testing Picture 1

Figure 16: Model Testing Picture 2

Figure 17: Recognising Model Testing Picture 3

Figure 18: Recognising Model Testing Picture 4

Figure 19: Recognising Model Testing Picture 5

4.2. Automated Log Sweep Calculation

The automatic log recognition model based on deep learning will calculate the sweep ratio of target logs by analysing the position of the pixel points. This step was achieved based on the result of the model. As shown in Figures 20 to 24, all the blue dots represent the pixel points of the mask. But it looks like many blue lines because the dots are so small. These blue areas are made up of dots which can be seen if the image is zoomed in. The red lines are the fitted lines of all these pixel dots, and the yellow lines are the midpoint lines at each end. The equation of the fitted line is shown in the bottom left corner. The fitted line is assumed to be a quadratic approach, and the model will automatically work out the complete functional equation. The midpoint line is straight. The sweep ratio of the log is automatically calculated using the sweep equation and is shown in the bottom left corner of the diagram.

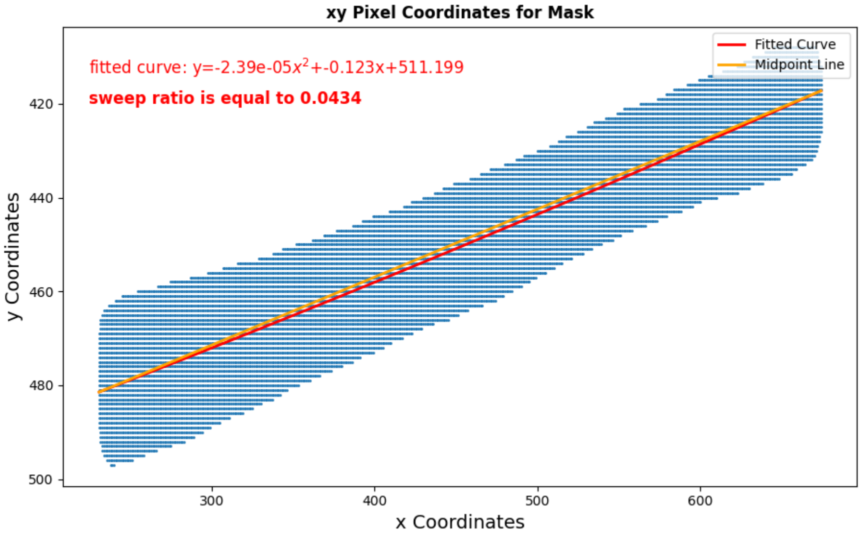

Figure 20: Sweep Ratio Calculation for Test Picture 1

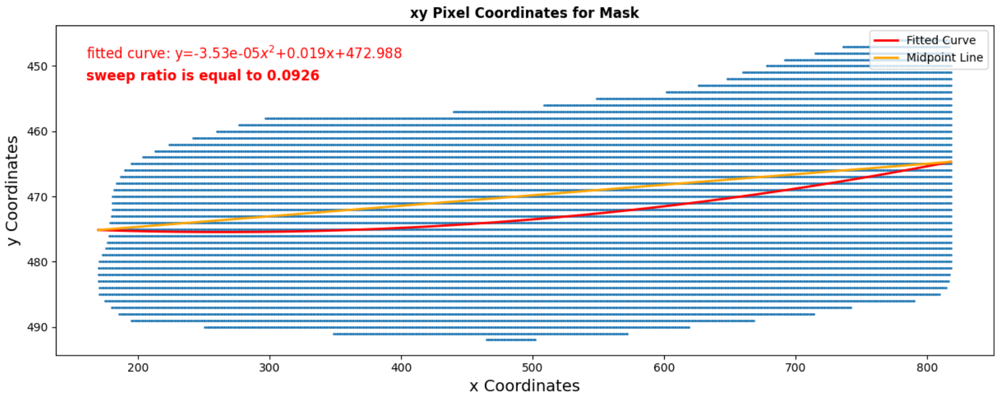

Figure 21: Sweep Ratio Calculation for Test Picture 2

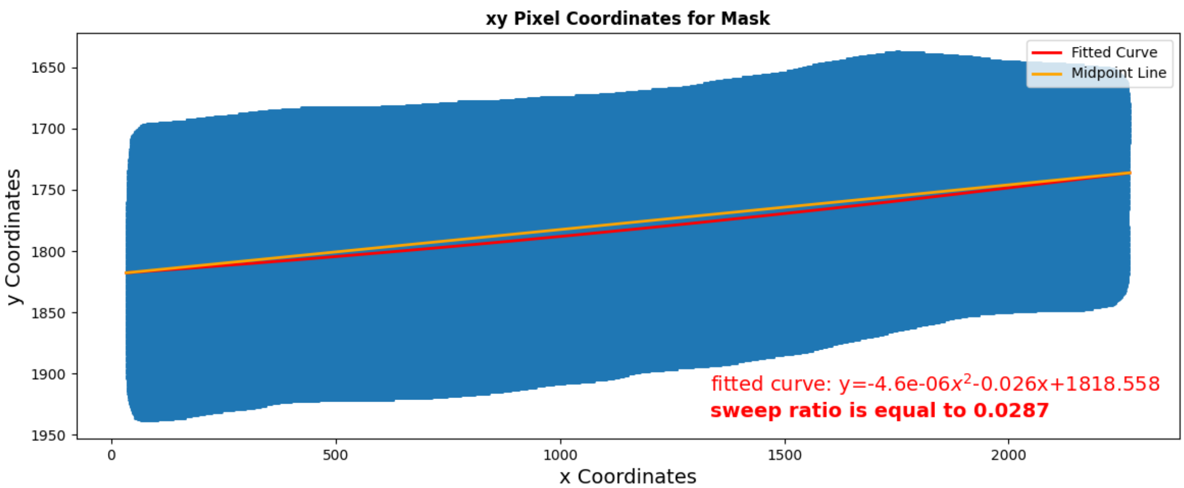

Figure 22: Sweep Ratio Calculation for Test Picture 3

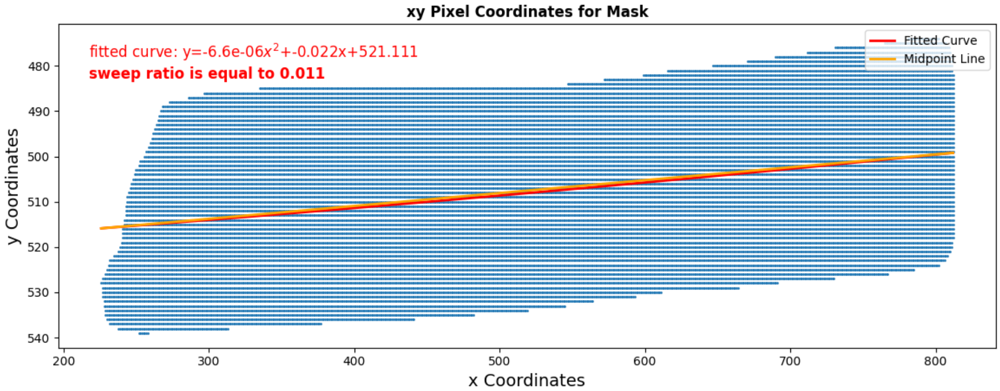

Figure 23: Sweep Ratio Calculation for Test Picture 4

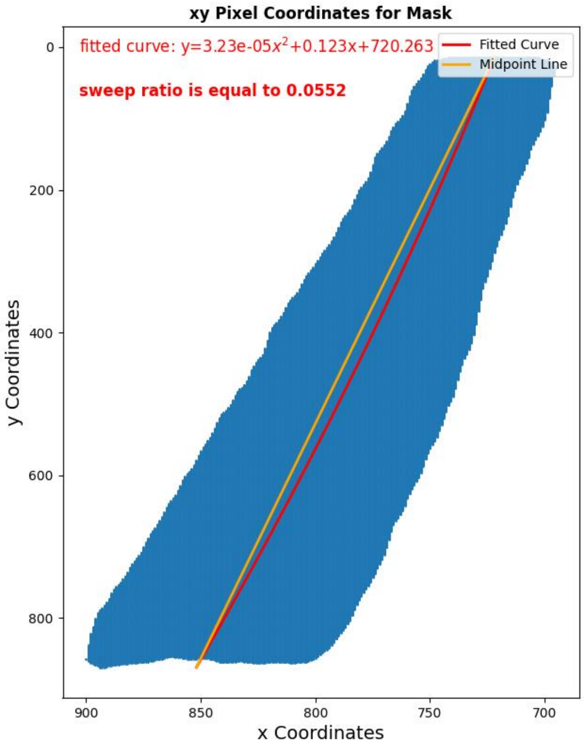

Figure 24: Sweep Ratio Calculation for Test Picture 5

The accuracy presented by the model is very high if one looks at the results of a few samples in the test set alone. But this may not mean that the model is very generalisable. The practicality of the model cannot be judged solely based on the accuracy of the sample results. In machine learning, Mean Average Precision is often used to judge how good a model is. It can be understood as an average of the various evaluated values of the model.

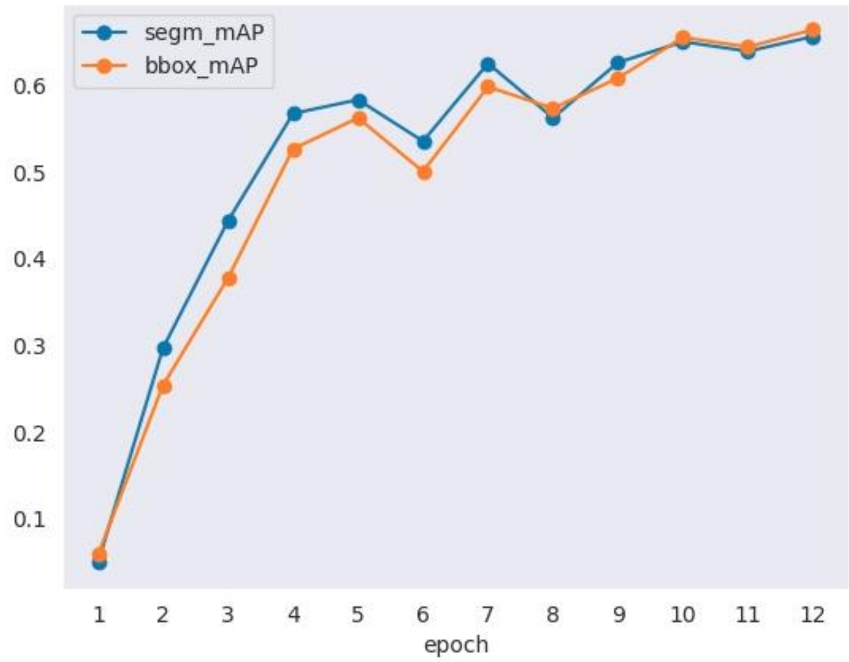

Figure 25: Mean Average Precision for Pre-trained Model

In Figure 25, the x-axis represents the epoch, where an epoch is a complete pass through all the training data. Y-axis represents the percentage of mAP. The blue line segment shows the data for the model mask, while the orange shows the value of the detection box. It can be seen that the model as a whole reached 65% mAP at the 12th epoch, which is much higher than the basic Mask R-CNN model and proves that the model produced by this project is relatively good.

Backbone	Style	Lr schd	Mem (GB)	Inf time (fps)	box AP	mask AP
R-50-FPN	caffe	2x	4.3		40.3	36.5
R-50-FPN	caffe	3x	4.3		40.8	37.0
R-50-FPN	pytorch	3x	4.1		40.9	37.1
R-101-FPN	caffe	3x	5.9		42.9	38.5
R-101-FPN	pytorch	3x	6.1		42.7	38.5
x101-32x4d-FPN	pytorch	3x	7.3		43.6	39.0
X-101-32x8d-FPN	pytorch	1x	-		43.6	39.0
X-101-32x8d-FPN	pytorch	3x	10.3		44.3	39.5
X-101-64x4d-FPN	pytorch	3x	10.4		44.5	39.7

Figure 26: Basic Mask R-CNN Model Average Precision (open-mmlab, 2021)

5. Discussion

5.1. Current Achievement

At this stage, the automatic identification of logs and sweep calculation is already possible at the picture stage. However, the overall accuracy can be improved, as the sample size has only 100 manually labelled images. The current method uses deep learning to train a model for log recognition. It assumes using only two cameras from a different direction. The optimal pictures should be obtained from both the top view and side camera views for getting the correct log sweep ratio.

5.2. Possible Improvement

Only the basic model has been made for this project, and there is still a lot of room for improvement. More work could also be done on the processing of the images themselves. For example, the data used for training could be enhanced: flip, rotation, scale, crop, translation, Gaussian noise, image brightness, saturation, and contrast changes. Also, when the target log is not perpendicular to the camera's position being designed to be placed. It will affect the final sweep ratio. This problem can be solved by using perspective transformation in computer vision, but this was not practised in this project.

There is another possible method to achieve the automated detection and sweep ratio calculation. It uses sensors in the processing head. When the head grabs the stem, it will mark the first point as the origin point. As the stem runs through the processor's head, the curve of the branch will be built into a model. Then the sweep percentage can be calculated automatically by the curvature method based on artificial intelligence. An example 3D wood scanner was built by LMI Technologies (chroma scan, 2021) and used in the sawmill. This sensor provides accurate, reliable data at production speed to optimise log and board breakdown decisions. It could profile lumber for size and shape, build high-resolution colour images to enable defect inspection, and conduct complete tracheid measurement for improved knot detection at the same time. However, it may not be suitable for the logging site operation due to the complex environment and other factors (Thies, 2004).

Another idea is to use high-definition cameras plus a visual recognition algorithm and then add LIDAR to ensure system stability. In autonomous vehicle driving, more and more manufacturers are choosing to include LIDAR to improve performance. In terms of detection accuracy, LIDAR has the advantages of high detection accuracy, wide detection range and high stability. However, due to the large amount of data acquired by LIDAR, high-performance processors are required to process the data. It is, therefore, more expensive but offers more assurance in terms of accuracy.

5.3. Future Consideration & Challenge

There are many possible options for the next direction. The current phase of the project could be packaged as an executable for others to test. Also, a live webcam for detection and computation could be implemented because the video combines many pictures frames. The calculation of lengths by deep learning was not included in this project, as the data used to train this detection model was obtained from videos on the internet. These data do not have a standard distance parameter (the distance from the camera to the logs).

In addition to this, there is a possibility for improvement in the overall accuracy of the model. As time goes on, more new algorithms appear or are updated (e.g., Yolo series, SDD, etc.). These new algorithms may require much less performance from the training hardware or significantly improve the detection accuracy. Since Mask R-CNN was introduced in 2017, four years ago, this might not be a relatively old concept in other fields. But in this aspect of deep learning, the rate at which model data is updated should not be underestimated. It will be an excellent choice to use a new and better quality base model to train the project's model.

6. Conclusion

Overall, the first phase of the entire project has been successfully realised. One hundred images were manually annotated and data transformed, and then the parameters were modified on top of the Mask R-CNN model for the training of the final model. After the training model was obtained, the new data were tested to obtain the results. From there, the test results can be analysed and extracted to obtain a pixel coordinate map. Finally, the calculation of the log sweep is completed by programming the fundamental mathematical model. Although there is still room for improvement in the overall accuracy, it is only a matter of collecting more data samples.

The next step in the project can be taken with a trained model and a method for extracting and analysing the resulting data. It is the testing of the video phase, the real-time computation, and the interfacing with actual log length data to complete the classification of the sweep specification. In addition to this, the existing model can be enhanced.

7. References

chroma scan. (2021). Retrieved from LMI Technologies: https://lmi3d.com/brand/chroma-scan/

Competenz. (2017). *Best Practice Guidelines for Manual Log Making and Processing.*

Emmert-Streib Frank, Y. Z. (2020). An Introductory Review of Deep Learning for Prediction Models With Big Data. *Frontiers in Artificial Intelligence*, 4.

Fengting Yang, Q. S. (2020). Superpixel Segmentation With Fully Convolutional Networks. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 13964-13973). IEEE/CVF.

Hamner, P. &. (2007). The frequency and level of sweep in mixed hardwood saw logs in the eastern United States. *Forest Products Journal*, 57.

He, H. (2019). The State of Machine Learning Frameworks in 2019. *The Gradient*.

He, K. G. (2017). Mask R-CNN. *IEEE international conference on computer vision.*, (pp. pp. 2961-2969).

He, K. Z. (2016). Identity mappings in deep residual networks. *European Conference on Computer Vision*, (pp. 630-645).

He, K. Z. (2016). Deep residual learning for image recognition. *IEEE conference on computer vision and pattern recognition*, (pp. 770-778).

Jiayun Wang, Y. C. (2020). Orthogonal Convolutional Neural Networks. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 11505-11515). IEEE/CVF.

Maclaren, J. (2000). How Much Wood Has Your Woodlot Got? *Forest Research Bulletin No. 217.* Rotorua: New Zealand Forest Research Institute Limited.

open-mmlab. (2021). *MMDetection*. Retrieved from GitHub: https://github.com/open-mmlab/mmdetection

Shaoqing Ren, K. H. (2016). *Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks.* arXiv.

Stanford University. (2020). *CS231n Convolutional Neural Networks for Visual Recognition.* Retrieved from https://cs231n.github.io/convolutional-networks/

Thies, M. (2004). Three-dimensional reconstruction of stems for assessment of taper, sweep and lean based on laser scanning of standing trees. *Scandinavian Journal of Forest Research*, 19(6), 571-581.

Thomas W. Edgar, D. O. (2017). Machine Learning. In *Research Methods for Cyber Security* (pp. 153-173).

Tsung-Yi Lin, P. D. (2017). *Feature Pyramid Networks for Object Detection.* New York: Cornell University.

Zhaojin Huang, L. H. (2019). Mask Scoring R-CNN. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 6409-6418). IEEE/CVF.

8. Appendix A

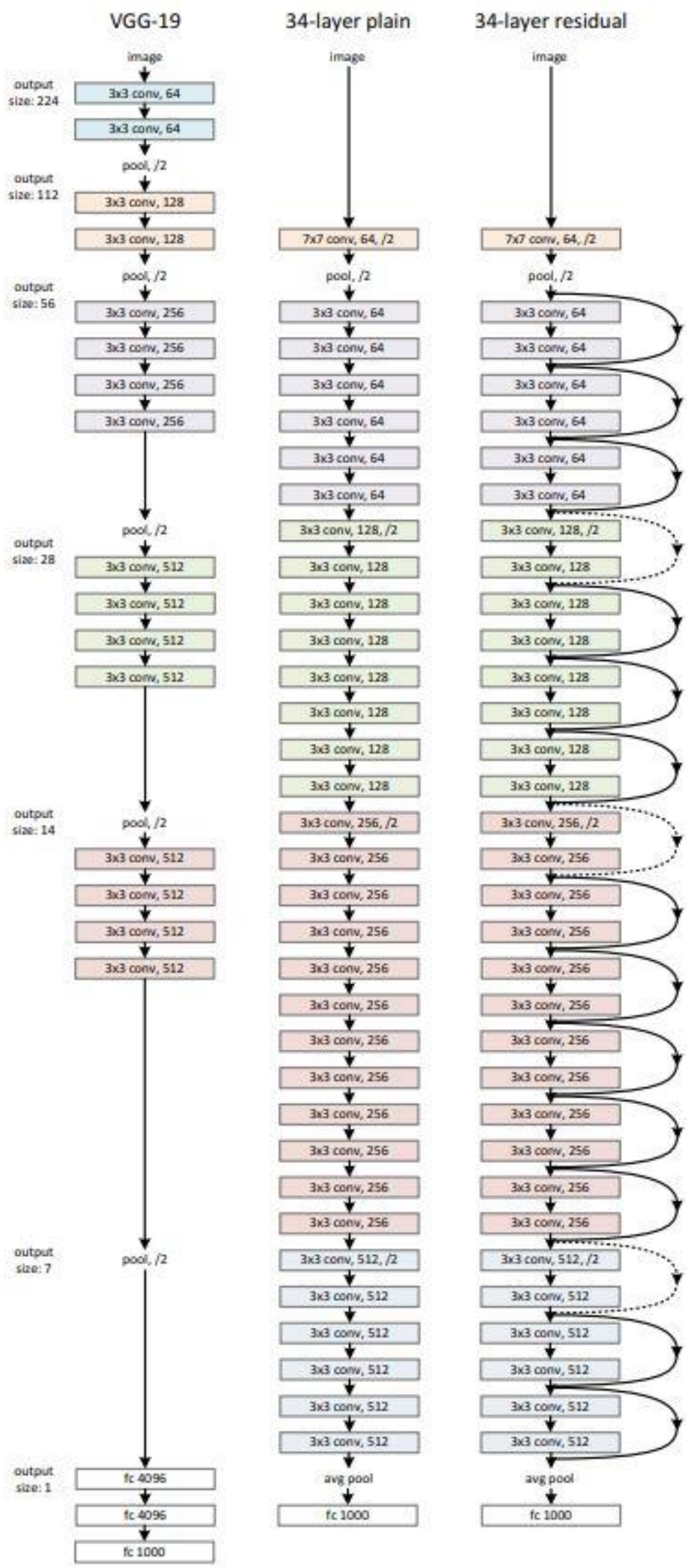

Figure 27: The Comparison between VGG Net and Res Net