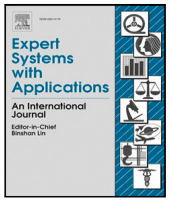
Expert Systems With Applications 246 (2024) 123117

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa



Convolutional neural networks for quality and species sorting of roundwood with image and numerical data

Julia Achatz [a], Mirko Lukovic [a], Simon Hilt [a], Thomas Lädrach [b], Mark Schubert [a,*]

[a] Empa Swiss Federal Laboratories for Materials Science and Technology, Cellulose and Wood Materials Laboratory, Überlandstrasse 129, Dübendorf, 8600, Zürich, Switzerland
[b] OLWO AG, Bollstrasse 68, Worb, 3076, Bern, Switzerland

ARTICLE INFO

Keywords:
Image classification
Computer vision
Roundwood sorting
Noisy real life dataset
Recommendation system
Process automation

ABSTRACT

Roundwood sorting is still a manual process in many Swiss sawmills, requiring employees to visually inspect and categorize thousands of logs per day. The heavy workload can be both physically and mentally taxing and can lead to increased rates of human error. State-of-the-art automation systems like X-ray log scanners are expensive and difficult to integrate into existing process lines. This paper proposes a novel recommendation system that leverages recent advances in image classification to automate roundwood classification by quality and species. The system integrates a camera to capture cross-sectional images of logs and record numerical data, such as length, taper, and diameter. The analysis of the resulting dataset highlights the challenges of data imbalance and noise, which makes classification difficult and, in some cases, impossible. However, by using selected datasets with reduced noise, state-of-the-art Convolutional Neural Networks (CNNs) can extract quality and species features. Quality models learn from a manually selected and simplified dataset, featuring samples that experts can clearly classify based on the image's information. Species models are trained on a label-noise-reduced dataset, reflecting real-world complexity. The accuracy on the selected dataset for three quality classes is 80%. The species determination is less challenging and reaches 91% accuracy on a synchronized dataset for the main species spruce and fir. Overall, this paper highlights the potential of Machine Learning in augmenting the roundwood sorting processes and presents a novel system that can improve the efficiency and accuracy of the process.

1. Introduction

Wood, as a foundational material, has profoundly influenced our homes, infrastructure, and culture from ancient civilizations to modern societies. With its exceptional weight-performance properties, sustainable availability, and significant CO_2 storage potential, wood stands as a crucial resource for the future. As demands and global competition continue to rise, the automation of processes in sawmills becomes increasingly important. Optimizing the sorting of roundwood at the beginning of the value chain ensures more efficient utilization of this valuable resource.

The roundwood sorting process is the initial step in conventional sawmills. It involves categorizing logs based on quality, species, and dimensions. Despite existing systems such as X-ray scanners or CT scanners, manual visual sorting is still prevalent in many sawmills (Niemz, Teischinger, & Sandberg, 2023). However, this highly demanding and labor-intensive task often leads to inaccurate sorting results that fluctuate significantly depending on the employee, load, and time of the day. Furthermore, the human eye already reaches its limits, preventing further increases in productivity.

Computer Vision algorithms, particularly Convolutional Neural Networks (CNNs), present a compelling solution for automating the roundwood sorting processes. Their ability to analyze visual data and extract meaningful information has revolutionized the way industrial tasks are performed. Therefore this paper proposes a novel recommendation system that captures images from roundwood logs and uses Computer Vision algorithms to make predictions. This system has the advantage of being cost-efficient and easy to integrate into existing process lines, which makes it attractive especially for small sawmills. On top, state-of-the-art Deep Neural Networks (DNNs) achieve high accuracy, are adaptable, scalable and have the ability for continuous improvement.

The first part of the paper introduces state-of-the-art image classification models and existing automated roundwood sorting systems. Section 3 provides background about the roundwood sorting process

* Corresponding author.
E-mail addresses: julia.achatz@empa.ch (J. Achatz), mirko.lukovic@wsl.ch (M. Lukovic), simon.hilt@empa.ch (S. Hilt), t.laedrach@olwo.ch (T. Lädrach), mark.schubert@empa.ch (M. Schubert).

https://doi.org/10.1016/j.eswa.2023.123117
Received 24 October 2023; Received in revised form 24 December 2023; Accepted 29 December 2023
Available online 4 January 2024

itself. The integration and implementation of the novel image classification system are described in 4. Finally Section 5 shows the results and concludes with a thorough evaluation. Potential future research is discussed in Section 6.

2. Related work

2.1. State-of-the-art image classification models

There are various methods to classify images using computer vision. Traditional approaches include manual feature engineering using statistical, structural and mathematical methods. For instance, Wang, Wang, Yu, and Li (2019) utilized gray-level co-occurrence matrices to distinguish wood species from boards, while Song, Li, Meng, Wu, and Cai (2017) employed a locally encoded transform feature histogram for classifying texture images. On the other hand Convolutional Neural Networks (CNNs) have gained significant attention in recent years and have been widely applied in various fields like image classification, object detection, and image segmentation. CNNs use filters which are applied via convolutions to extract features from images. These features are then classified by different Machine Learning (ML) approaches. CNNs eliminate the need for manual feature extraction in traditional computer vision, offering an end-to-end learning approach. Deep Learning approaches are proven to achieve higher accuracy on image classification tasks, require less expert knowledge and provide a high flexibility (O'Mahony et al., 2020).

There exists many state-of-the-art CNN architectures reaching considerably high accuracy on the large scale visual recognition challenge (ILSVRC). Noteworthy advancements include VGG architecture, boosting performance while reducing parameters through multiple convolutional layers using compact 3×3 filters (Simonyan & Zisserman, 2014). One of the most popular architectures for image classification is ResNet (Residual Network) (Kaiming, Xiangyu, Shaoqing, & Jian, 2015). It uses a Deep Residual Network to address the problem of vanishing gradients. The residual connections allow information to flow directly through the network, enabling the training of very deep networks. InceptionNet is another renowned image classification model (Szegedy et al., 2014), which applies a combination of 1×1, 3×3, and 5×5 convolutions to extract features at multiple scales and concatenate them to generate the final output. Another emerging model is called EfficientNet (Tan & Le, 2019). It is known for its exceptional performance and efficiency by effectively balancing model size, accuracy, and computational resources. EfficientNet achieves this balance by using a compound scaling technique that optimizes the model's depth, width, and resolution. There are also many other popular image classification models like DenseNet (Huang, Liu, Van Der Maaten, & Weinberger, 2017), MobileNet (Howard et al., 2017), and AlexNet (Krizhevsky, Sutskever, & Hinton, 2012). Chen et al. (2021) provides a comprehensive overview of state-of-the-art CNN architectures.

To choose an appropriate architecture, we considered task-specific requirements like computational power, prediction times, image diversity, and pre-trained network availability for transfer learning. To demonstrate our approach, we opted for well-established network – VGG, ResNet, EfficientNet, and InceptionNet – which, with their proven performance, effectively serve our purpose. While other architectures may yield similar results, our selection aligns precisely with our criteria, showcasing the potential success of our new approach.

2.2. State-of-the-art roundwood sorting

If we go back in time, the first timber businesses were probably based on a simple agreement regarding the number of logs. Over time, manual methods evolved to determine the log volume by measuring the diameter and length. In 1960 sawmills started to apply automated log scaling and sorting techniques (Niemz et al., 2023). Nowadays there are many different techniques on the market to automatically scale and grade logs and new techniques are constantly evolving.

Ultrasound-based technologies, radiography, and X-ray-based systems can measure the internal characteristics of a log. In 2011 the company Microtec[1] presented the first high-speed computer tomography (CT) log scanner. The accuracy of an X-ray log scanner when classifying into three different quality classes is approximately 85%, which is better than achieved with optical or manual visual judgment (Niemz et al., 2023). Microtec also uses AI to support the CT log scanner in assessing wood defects such as dead branches, resin pockets, or compression wood. The disadvantage of X-ray and CT scanners is that they are more expensive, less scalable, and less easy to integrate into existing process lines, compared to our lightweight image classification approach.

3. Roundwood sorting process

The roundwood sorting process involves several steps, illustrated in the lower part of Fig. 5. Initially, an infeed conveyor (1, 2) separates the delivered logs, which then pass through a detector capable of measuring length, diameter, and detecting metal (3). Next, the logs are scaled to uniform length (4). The cut can also be done to facilitate quality determination. Subsequently, they proceed through a house where experts assess the quality and species of the logs (5).

The sorting process serves two main purposes: determining the appropriate price for the timber vendor and achieving improved process control and production planning. Such classification results in grouping logs together based on their suitability for being sawn using the same saw-line settings. The successful implementation of a sorting system has a significant impact on both the value yield and the overall production capacity (Niemz et al., 2023).

According to the guidelines outlined in "Schweizer Handelsgebräuche für Rohholz" (Bundesamt für Umwelt BAFU, 2010), softwood logs are typically sorted into four distinct quality classes: A, B, C, and D. Logs classified as grade A exhibit excellent quality, being virtually free from defects and possessing only minor imperfections. Grade B strains represent good to moderate quality, characterized by a lack of heavy knots or coarse features, though slight defects may be present. Strains with quality C fall within the medium to below average quality range and exhibit significant defects. Quality D wood, by its inherent characteristics, cannot be included in the other classes. Furthermore, there are staple strains (Klammerstämme) that exhibit different qualities on different parts of the same log. In our dataset, these are classified into classes AB or BC. Additionally, class C strains are reassigned to class Kae when infested with beetles, while class Me includes wood with metal contaminants. The models in this paper primarily focus on the quality classes B, C, and D, which occur most frequently.

Roundwood classification is based on well-defined characteristics. Table 1 shows permissible features for classes B, C, and D for spruce and fir.

Most of the features can be detected by the eye either on the stem or the cross-section. On the cross-section, several key features that are crucial to learning include average annual ring width, reaction wood, cracks, ring dish, pitch pockets, and discoloration. These features can be accurately measured and calculated using the formulas provided in Fig. A.18 in Appendix A. The remaining features, predominantly observed on the stem itself, encompass fused and non-fused branches, curvature, and spiral growth. Corresponding formulas are presented in Fig. A.19 in Appendix A.

The most important numeric feature is the so-called taper, which refers to the gradual reduction in trunk thickness from the base to the crown. Therefore three different diameters are measured, D1 (root),

[1] https://www.microtec.eu/.

Table 1
Overview of key characteristics for various quality categories B, C, and D for spruce and fir species (modified from Bundesamt für Umwelt BAFU (2010)). Visually assessable features are indicated in light blue, numerical attributes in dark blue. Green markings represent visual information visible on the cross-section, while red markings indicate features on the stem.

Features/Quality	B	C	D
Fused branches	<4cm	< 6cm	allowed
Non-fused branches	<3cm	<6cm	allowed
Curvature	<1cm/m	<1cm/m	<2cm/m
Taper D2 < 20cm 20cm < D2 < 35cm >= 35cm	 < 1.0cm/m < 1.5cm/m < 2.0cm/m	 < 2.0cm/m < 2.5cm/m < 3.0cm/m	 < 2.5cm/m < 3.0cm/m < 4.0cm/m
Spiral growth	< 4cm/m	< 6cm/m	unlimited
Average annual ring width	<6mm	unlimited	unlimited
Whine growth (fir)	< 1m	allowed	allowed
Reaction wood	<10%	< 25%	allowed
Cracks from the middle	< 35% of D2	< 50% of D2	allowed
Ring dish	not allowed	< 20% of D2	< 35% of D2
Pitch pockets	not allowed	> 2cm, max. 5	allowed
Insect damage	not allowed	light infestation allowed	allowed
Discolourization	not allowed	allowed	allowed

Part of the tree: Cross Section Stem Type of information: Visual Numeric

D2 (mid), and D3 (bottom). The taper t is then calculated with the following formula, where l is the length of the stem:

$$t = \frac{D1 - D3}{l} \text{ [cm/m]} \quad (1)$$

A higher taper value (depending on D2) has a detrimental effect on the quality.

The expression of characteristics/features can vary depending on the location. Our partner sawmill OLWO,[2] for instance, experiences a prevalence of reaction wood due to its location in an area with numerous slopes and exposures. Additionally, resin pockets are frequently encountered.

It is worth mentioning that there exists a concept called discretion in the quality assessment, as outlined in paragraph 2.2.3 of "Schweizer Handelsgebräuche für Rohholz" (Bundesamt für Umwelt BAFU, 2010). This provision allows sporadic shortcomings to be balanced by superior qualities. But, if these shortcomings accumulate excessively, they could lead to a lower quality classification. Some features are only important for distinct species, e.g. eccentric growth is only considered an important feature for douglas fir, pine, and larch.

Overall, the sorting process is a complex task influenced by numerous factors. It is essential to consider the high variability of wood, which significantly complicates the automatic classification of roundwood.

4. Materials and methods

The main idea behind our research, is to use Convolutional Neural Networks to predict the quality and species of logs based on images and numerical data. The development of the roundwood classification method comprises six iterative phases, which are visualized in Fig. 1: Data acquisition, analysis, cleaning, preprocessing as well as model training and deployment. Subsequent chapters will provide an in-depth exploration of each of these phases.

[2] https://www.olwo.ch/de/.

4.1. Data acquisition

The initial phase is the data acquisition, as depicted in the lower part of Fig. 5. Hence, we integrated a camera system into the production line at our partner sawmill, OLWO. While pushing the tree trunk through position 1, the system detects the distance to the front end of the stem. The position of the camera is adjusted in x-direction accordingly to ensure a consistent distance between the camera and the stem. When the tree trunk has reached position 2, the trigger signal for the camera is generated and a pixel image is acquired. The file name of the image is always provided with a consecutive, unique number, date (dd,mm,yyyy), and time (h:m:s:ms). At position 5 the operator enters a quality and species decision, which is logged as the label of the image. The metadata measured at position 3 is also saved.

4.2. Data analysis

The resulting data undergoes further analysis through a series of experiments and studies. Section 5 provides further information on the dataset size and class distributions. Furthermore, three different noise sources are analyzed: image noise, direct and indirect label noise (see Fig. 7). Image noise is caused by variations in intensity levels and loss of sharpness, affecting the clarity and quality of visual information. On the other hand, there is direct label noise, which means that the label assigned to an image does not align with the ground truth of the sample. This can be due to human uncertainty or asynchronization. Asynchronization means that labels and images get shifted with respect to each other in the production timeline. Reasons can be images with either two stems or no stem on them, connection problems, or system reboots. This results in labeling an image with the label of the previous or following image, which then propagates over time. The last kind of noise, indirect label noise, is caused by missing information in the data, which can occur either due to information being hidden/occluded (e.g. by dirt, or color) or not captured in the image. In particular, there is a lack of information about features on the stem (see Table 1) since the current dataset only contains cross-section images. Furthermore, it is possible that certain characteristics, such as rot or discoloration, disappear after cutting, or new features, such as resin pockets, become visible.

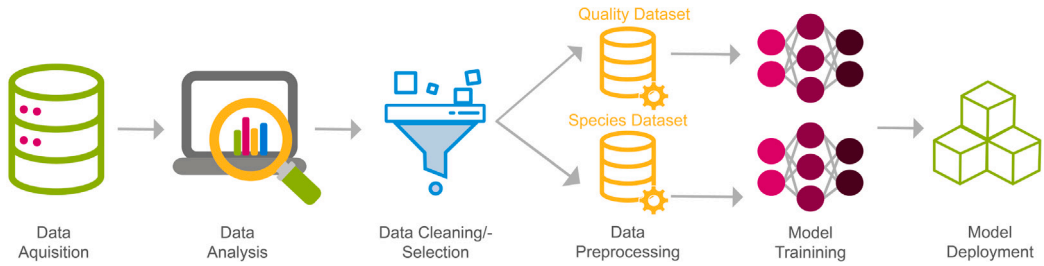


Fig. 1. ML-Pipeline: Overview of the development process.

For each of the noise sources, some experiments are carried out to determine the number of affected samples. For analyzing the image noise, 500 random samples are scrutinized for overexposure and blurring. In order to analyze the indirect label noise, 1478 stems were observed over a period of 2 days. For each stem, it was observed if there were important features on the stem that became visible/neglected after scaling at position 4. The last study examines the synchronization problem over a period of 16 days and thus the direct label noise. Since for technical reasons it was not possible to check at which time of the day the asynchronization took place, an average corruption rate of 50% was assumed. With that, an estimation of the percentage of affected labels can be derived. Through these systematic analyses, valuable insights into the dataset's characteristics and noise sources are obtained. These findings contribute to a comprehensive understanding of the challenges and potentials associated with the dataset.

4.3. Data cleaning and selection

Due to the high noise content (see Section 5.1), especially label noise, models are not able to learn from the entire dataset. Since it is not possible to filter out all affected samples, two separate, noise-reduced datasets were created, one for training the species and one for the quality model. The species models learn from an asynchronization-free dataset, ensuring that each label matches the image. It is collected during the 16 days of the study mentioned in Section 4.2. This dataset is label noise reduced but retains the complexity of the original data set. It contains 6577 samples and is further referred to as species dataset.

Quality models learn from a manually selected and simplified dataset, featuring samples that experts can clearly classify based on the cross-sectional image information. This reduces the complexity as well as the noise in the dataset. We further call this dataset a quality dataset. The overall quality dataset includes 1800 samples (600 in each class).

The real-time prediction requires the recognition of image noise including overexposure and blurriness during the image preprocessing. For the detection of the blurry images, the suggested system uses the variance of the Laplacian with a threshold of $\theta = 10$. The idea is to convolve one channel of an image with the Laplacian 3×3 kernel (Rosenbrock, 2015):

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \tag{2}$$

To get the amount of blurriness, one can take the variance of the response. If the variance falls below the threshold θ in this case, the image is classified as blurry. In more detail, the Laplacian operator measures the second derivative of an image, which means that it highlights regions with a rapid intensity change, which in the end detects edges. The lower the variance the fewer edges are present in the image and the higher the probability of having a blurry image.

To accurately identify overexposed images, an algorithm utilizes a pixel analysis approach. This involves evaluating the proportion of bright pixels within the cross-sectional part of the image. The algorithm employs two experimentally determined thresholds. The first threshold is used to classify pixels as excessively bright, while the second threshold establishes the desired proportion of such pixels within the cross-section. The first one is set to 245 and the second one to 0.34 based on experiments. Both detectors are tested on a dataset with 500 randomly selected samples and compared to manual determination of blurriness and overexposure. The number of blurry images was artificially expanded to be able to make better statements.

4.4. Data preprocessing

Prior to training the models, a customized preprocessing pipeline performs several steps to generate a suitable tensorflow (tf) input dataset for training. Fig. 3 shows an overview of the preprocessing pipeline. The initial step in the pipeline is the segmentation of the cross-sectional area to remove the background. This unifies the input images and helps the final model to find relevant features. This is helpful due to the high variability of the data (different sizes, exposure, etc.) and the small size of the dataset. To accomplish this, the paper proposes employing a UNet (Ronneberger, Fischer, & Brox, 2015) image segmentation model pre-trained on MobileNetV2 (Howard et al., 2017). The model is trained on a dataset containing 154 manually labeled images and tested on a dataset with 38 labeled images. In order to remove isolated pixels which are still left in the background three morphological transformations are carried out: Opening (erosion followed by dilation) followed by two times closing (dilation followed by erosion) with different kernel sizes (5×5, 20×20, 2×2) each. By identifying the border pixels of the segmentation mask, the images are subsequently cropped and resized with a pad to 1024×1024 pixels. Fig. 2 shows some example images for classes B, C, and D with segmented backgrounds.

The next steps are carried out during runtime/training of the model. After reading the data to the memory, image and numerical data are shuffled by random permutation. The quality and wood species labels are one-hot encoded to make them machine-readable. After that, the dataset is split into training (80%) and validation dataset (20%). The training dataset is balanced by over-, under-sampling, or re-weighting if necessary, followed by another random shuffling. After casting the dataset to a tf dataset, the images are preprocessed including parsing and normalizing. The normalization/scaling depends on the method used in the ImageNet pre-trained network. Simple augmentation like flipping and rotation can be applied optionally. For the numerical data, a standard scaling (sklearn standard scaler) is used. In the end, the dataset is batched for training the model.

4.5. Model building and training

After preprocessing the data, several different models are built, to predict either quality or species. Therefore four different state-of-the-art models are used: ResNet50, EfficientNetB0, VGG19 and InceptionNetV3. Utilizing the Keras framework, all networks are built upon pre-trained models from the ImageNet dataset. The initial layers (165 for ResNet50, 100 for EfficientNetB0 and InceptionNetV3, and 10 for VGG19) are kept frozen, while the remaining layers are fine-tuned on the roundwood dataset. The architecture remains consistent across all four models, featuring an AveragePooling2D layer, followed by Flatten,

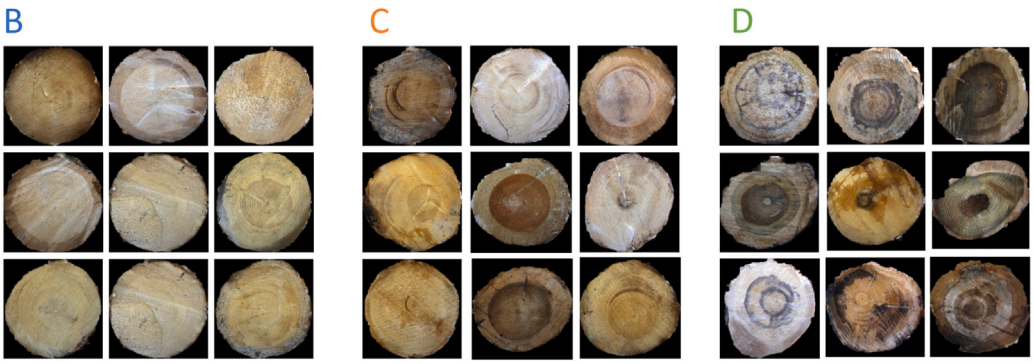


Fig. 2. Example images for each quality class B, C and D after image segmentation.

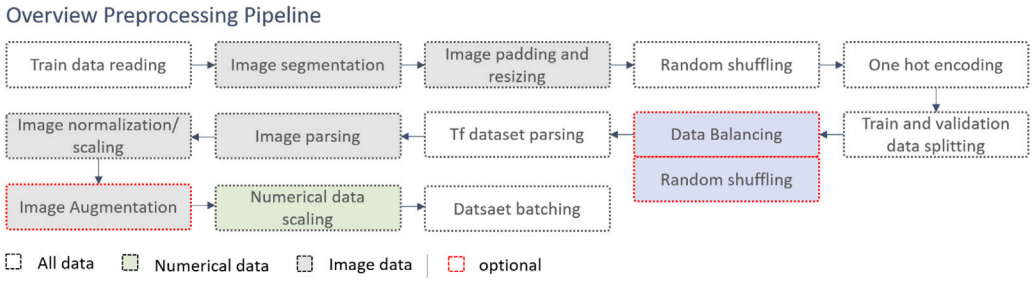


Fig. 3. Customized preprocessing pipeline used for all models.

a Dense layer with ReLU activation containing 128 neurons, and a Dropout layer. This is succeeded by a final dense layer with a softmax activation function, consisting of 3 neurons for quality (B, C, and D) and 2 for species (spruce and fir). For the loss function, categorical cross entropy is used together with an Adam optimizer. To prevent the model from overfitting, the early stopping method is used with patience of 5 with a maximum of 300 epochs. On top dropout regularization and the L2-regularization in the last layer help to generalize well on unseen data. All manually tuned hyperparameters can be found in Appendix B.

A mixed model includes numerical data, mainly taper, which is important for quality prediction. This model gets the image and the numerical data as input, combines both in a concatenate layer, and makes a combined prediction. Fig. 4 shows the general architecture used for all models including the layers needed for the MixedModel. In addition, numerical data is used as a hard constraint (based on Table 1). This means that after getting the prediction from the image classification results are downgraded in case of a too-high taper value (see Table 1).

The proposed models were trained with keras based on tensorflow platform. The used Python version is 3.7. The experiments were done on an AMD Ryzen Threadripper 3970X CPU with 3.70 GHz and 32-Cores and the models are trained on ASUS GeForce RTX 2080 Ti TURBO (11 GB, high-end) GPUs.

The species models were trained as binary classifiers for the classes spruce and fir. The quality models focus on the three quality classes B, C, and D mentioned above. For hyperparameter tuning a stratified 5-fold-cross validation is carried out. The final model is trained on the whole training dataset and is tested with a separated, stratified test dataset, which includes 20% of the overall dataset. Since the quality dataset is already balanced, no extra treatment for imbalance is necessary. For the species dataset reweighting is used.

4.6. Model deployment

Enabling real-time utilization of the recommendation system involves a series of distinct steps. Initially, the uploaded image undergoes parsing and error verification. In case of a damaged image file, manual intervention is necessary. After segmenting, the image is assessed for blurriness or overexposure (as detailed in Section 4.3). In case of a positive result, an alert is issued to prompt manual intervention by the sorter. The software then addresses potential synchronization problems within the image, such as dual stems or the absence of a stem. In instances of dual stems, the image is split into separate entities, while an empty image is excluded from consideration. After sequentially propagating the images through the models, the uncertainty of the prediction can be evaluated further. This was not examined in more detail in the paper. Fig. 5 shows how the models predict species and quality during the running process.

5. Results and discussion

This section analyzes the dataset, shows the results of the preprocessing and the models, and evaluates them. It involves not only assessing model accuracy but also analyzing training and prediction times. Using this information, the system throughput can be calculated and compared to the current one. Additionally, it shows the potential impact of incorporating more data and higher image resolutions.

5.1. Results and evaluation of data analysis

The overall dataset contains 54.594 samples where 97.15% (53.039) of them are either of class B, C, or D. 43% (23.058 samples) of them are of class B, 50% (26.317 samples) of class C and 7% (3644) are of class D. Besides classes B, C and D, there are also 0.004% (2) samples in class AB, 0.2% (82) in class BC, 2.3% (1306) in class Kae and 0.3% (165) in class Me. 99.7% (54.438) of all samples are either of species spruce or fir. 79.9% (43.483) of them are spruces and 20.1% (10.955) firs. Only 0.08% (46) samples are of class douglas fir, 0.06% (32) of class pine, and 0.13% of class larch. Fig. 6 shows a statistical overview of the distribution of samples across quality and species classes. This barplot highlights the class distribution imbalance. Notably, classes B and C exhibit substantially higher representation compared to class D, with the remaining categories being present in mere fractional quantities. This pattern is also evident in the case of the spruce and

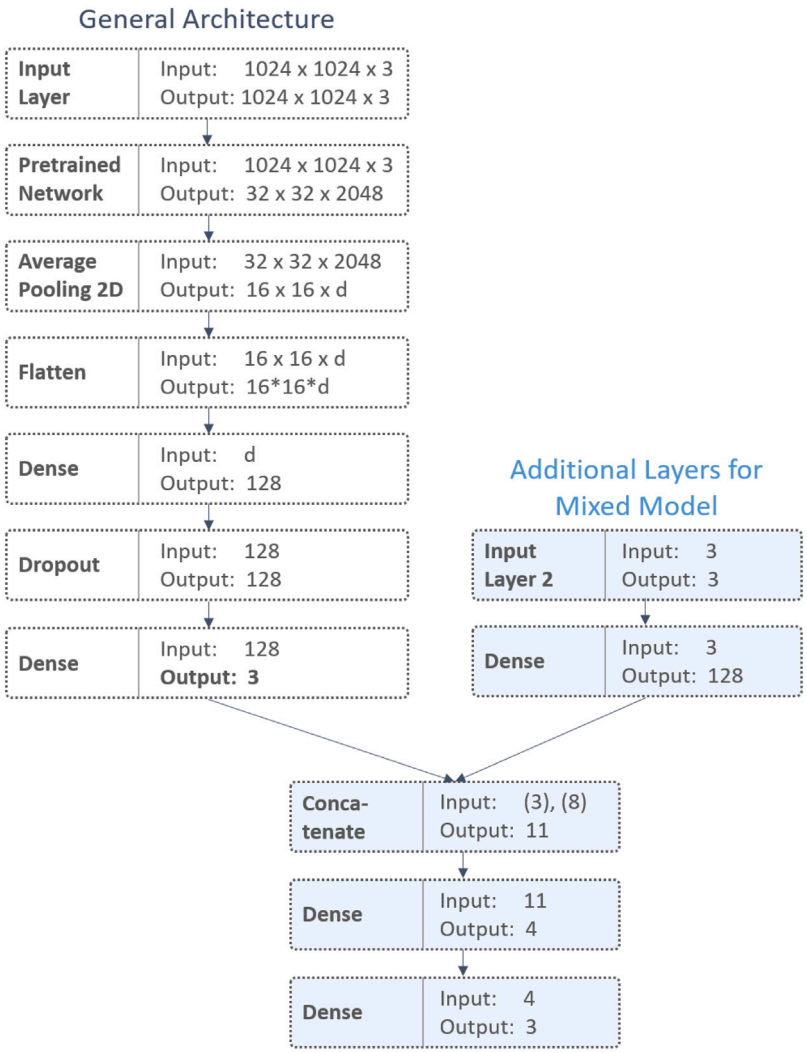


Fig. 4. General architecture of quality models. Pretrained network either stands for ResNet50, EfficientNetB0, VGG19 or InceptionNetB0. Layers colored in blue belong to the MixedModel, white marked layers belong to the general architecture. Values marked with d are different and depend on the pretrained network ($d_{ResNet50} = 2048$, $d_{InceptionNetB0} = 2048$; $d_{EfficientNetB0} = 1280$; $d_{VGG19} = 51$). Input for mixed models are the numerical parameters D1, D3 and length. Species models have the same architecture with the last dense layer having only 2 neurons.

fir classes, which enjoy significantly higher occurrence rates than the other species. Regrettably, the available data for all other species is insufficient to adequately train meaningful models.

Next, different sources of noise are studied in more detail. Fig. 7 shows the different sources of noise described in 4.1, including the results of the studies and possible solutions. The image noise is 7.4% with 6.4% of the images being overexposed and 1% being blurry. Both noise sources significantly reduce the information content in the image and thus can negatively affect the predictive power of the models. In addition, the strongly varying light and quality conditions can affect the model, which is not further investigated in this context. On top 34% of the 500 tested images are sprayed with color, partially making features less visible. Affected images need to be identified in advance to give warnings for manual intervention in the real-life process.

The occlusion noise affects 34% of the images. In 20% of the cases, features were no longer visible after the cut (often superficial rot) and in 14% of the cases, features became visible after the cut (often because features were previously covered by dirt or pitch pockets appeared). In general 98% of all stems are cut either for scaling or quality decisions. The information gap noise affects 25% of the images. In most cases, the missing information is related to non-fused or fused branches (19%). In 6% of cases, tree trunks are curved, discolored, or have spiral growth. It is important to note that affected samples do not necessarily have incorrect labels. For instance, in case the information from fused branches is missing but the cross-section still exhibits features pointing to the same class, there is no label noise for that particular sample. Furthermore, in the case of cuts, new features can emerge/disappear subsequently, without impacting the quality assessment decision. Possible solutions would be to add a second image from the stem and reposition the camera after the cut (position 4). To ensure coverage of all possible important features, each part of the trunk would have to be seen in one image, whereby it can be assumed that two images cover enough features to make a meaningful quality and species decision.

The direct label noise comprises more than 15% due to human uncertainty (Niemz et al., 2023) and 19% due to asynchronization. However, it is important to note that the 19% only applies to images affected by synchronization issues and does not necessarily imply incorrect labeling. 44% of the 19% asynchronized labels coincide with the correct label by chance, resulting in 10.6% of mislabeled samples (see Appendix D.1). Direct label noise needs to be disregarded during the training phase of a model and improved during real-time and testing.

Moreover, it is crucial to understand that the same image can be affected by more than one noise source at the same time, which is why the probabilities cannot be added. To estimate the overall label noise, one can use the inclusion–exclusion principle (Björklund, Husfeldt, &

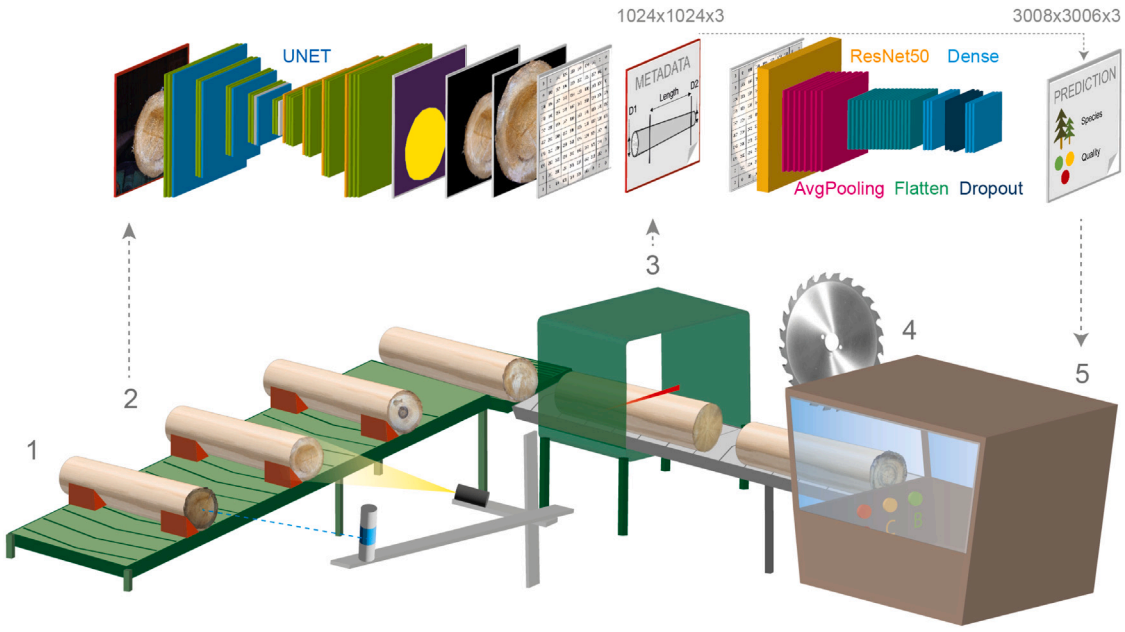


Fig. 5. Schematic illustration of the roundwood sorting process including the Machine Learning (ML) recommendation system. First the travel distance for the camera is calculated and the displacement is tracked for synchronization. Next the trigger signal for the camera is generated and picture is taken. At position 3 a scanner measures length and diameter. In position 4 the stems are cut to the appropriate length. In the last step, the operator enters his quality and species decision. The upper half of the image shows the Machine Learning process running synchronously. After preprocessing the image, predictions are made by state-of-the-art CNNs like ResNet, followed by Pooling, Dropout and Dense layers.

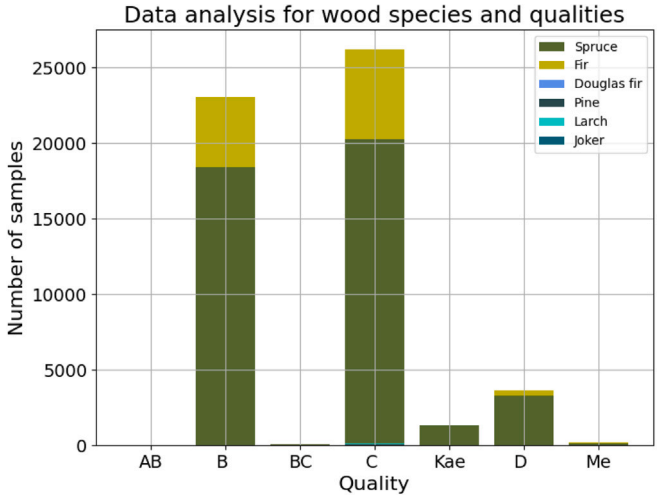


Fig. 6. Data analysis for wood species and qualities. It shows the number of samples in each of the seven different quality classes. The colors show different species within each quality class.

Koivisto, 2009). Since there are no fixed percentages of wrong labels for occlusion noise and information gap noise, a range of possible label noise is calculated. First, we assume that occlusion and information gap noise is 0. Next, we assume that all affected samples also have a wrong label. With that, we calculated a label noise range between 24% and 62% (see Appendix D.2). This shows that there is a significant amount of label noise that needs to be handled in the future. The first goal is it to minimize the noise as much as possible by incorporating a second image of the stem and addressing asynchronization with hardware and software (see 4.6). To handle the remaining label noise, the literature suggests different methods. Song, Kim, Park, Shin, and Lee (2022) provides an overview of existing label noise-handling techniques.

In summary, we have faced common real-world data challenges, which are also stated in the literature. Liu and Panagiotakos (2022) for example described real-world data as noisy, heterogeneous, incomplete, and unbalanced. Others also mention the problem of imbalance (Kotsiantis, Kanellopoulos, Pintelas, et al., 2006) and data quality in real-life data for smart manufacturing (Wang, Ma, Zhang, Gao, & Wu, 2018; Xu et al., 2022).

5.2. Results of data preprocessing/image segmentation

Fig. 8 shows the loss and accuracy of the Unet segmentation model over 30 epochs for training and validation dataset. The final train accuracy is 0.98 with a loss of 0.05, while the validation accuracy is 0.98 with a loss of 0.07.

The blur detector reaches an F1 score of 0.84 while the overexposed detector has an F1 score of 0.87. Appendix D.3 shows the confusion matrices for both detectors. Carefully tuning the thresholds for both detectors could improve the accuracy/F1 score. An alternative would be to train a neural network to detect overexposure and blurriness. Although this could lead to higher accuracy, this approach requires significantly more computational power.

5.3. Results of model training

This section shows the results of the quality and species models. The manually tuned hyper-parameters which are used for training are listed in Appendix B. Table 2 shows an overview of the results of the quality models. ResNet50, EfficientNetB0, VGG19 and InceptionV3 networks were trained respectively with and without hard numerical constraints, which is downgrading quality in case of a too high taper value (see Section 4.5) as well as a corresponding mixed input models.

Fig. 9 shows the confusion matrices of the final quality ResNet50 model without (a) and with (b) hard numerical constraint. The model reaches an accuracy of 78% and an F1 score of 0.79. Adding the numerical constraint slightly improves the accuracy mainly by downgrading B-classified samples to class C. EfficientNetB0 also reaches an accuracy of 78% and an F1 score of 0.78. Fig. 10 shows the confusion matrices of EfficientNetB0 without (a) and with (b) numerical constraint. The accuracy again increases after applying the numerical constraint to an accuracy of 80% and an F1 score of 0.8.

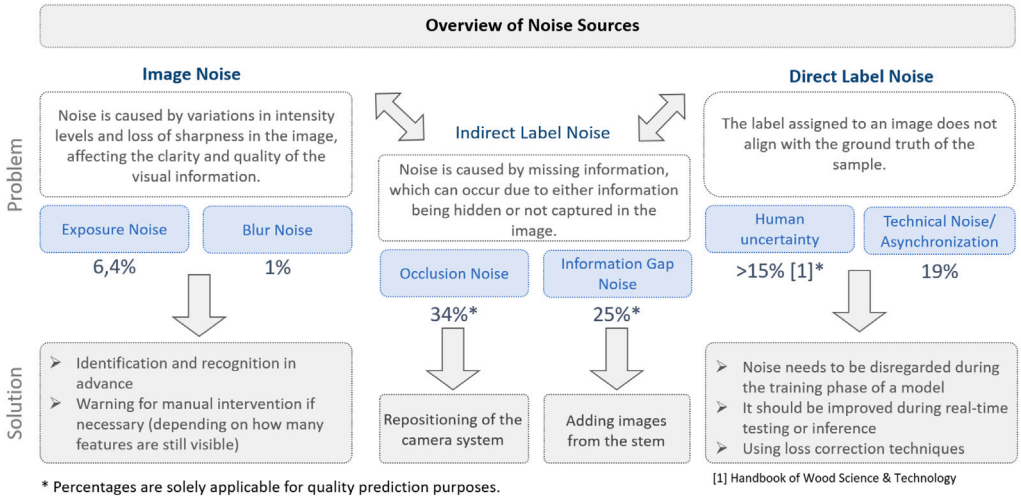


Fig. 7. Overview of different noise categories including the estimated number of affected samples based on studies. The main noise sources are image noise and direct and indirect label noise.

Table 2
Results of quality models.

Model	Avg. val acc/loss	Avg. val F1 score	Test acc/loss	Test F1 score
ResNet50	0.78 (±0.03)/ 0.84 (±0.09)	0.78 (±0.03)	0.78/0.76	0.79
Constraint	–	–	0.79	0.79
Mixed	0.58 (±0.05)/ 1.69 (±0.54)	0.56 (±0.07)	0.52/1.03	0.44
EfficientNetB0	0.79 (±0.01)/ 0.93 (±0.12)	0.79 (±0.02)	0.78/0.99	0.78
Constraint	–	–	0.80	0.80
Mixed	0.62 (±0.11)/ 0.91 (±0.08)	0.58 (±0.13)	0.56/0.98	0.56
VGG19	0.75 (±0.03)/ 1.43 (±0.17)	0.75 (±0.02)	0.63/2.45	0.61
Constraint	–	–	0.68	0.67
Mixed	0.43 (±0.07)/ 1.06 (±0.03)	0.42 (±0.07)	0.64/0.95	0.63
InceptionNetV3	0.63 (±0.07)/ 2.21 (±0.33)	0.62 (±0.05)	0.68/2.02	0.68
Constraint	–	–	0.68	0.69
Mixed	0.65 (±0.05)/ 2.21 (±0.33)	0.65 (±0.05)	0.71/1.91	0.71

The matrices show that EfficientNetB0 performs worse on class C but better on class B and D than ResNet50. In the future, ensemble methods may be used to achieve better accuracy. Rokach (2009) provides a review of different strategies to combine models of the same or different families.

Fig. 11 shows the accuracy (a) and loss (b) plot of ResNet50, which achieved the highest accuracy without numerical constraint. It shows clearly that the model is still overfitting due to the lack of data. VGG19 and InceptionNetB0 exhibit poorer performance compared to the other two models. The introduction of a numerical constraint for taper improves the accuracy/F1 score in all four models however, this enhancement is most pronounced in the case of VGG19. Among the mixed models, InceptionNetB0 performs best with a final accuracy and F1 score of 71%/0.71. Anticipating greater strides in quality predictions from numerical data is more likely in datasets with real-world complexity. In the present dataset, only images where the cross-sectional view already provides clear indicators of quality were selected. Consequently, substantial improvements from the inclusion of numerical data are less likely to materialize.

Existing literature primarily focuses on wooden board quality assessment and wood defect detection. For example, Affonso, Rossi, Vieira, de Leon Ferreira, et al. (2017) achieved 82.11% accuracy in categorizing wooden boards into three quality levels, and Nurthohari, Murti, and Setianingsih (2019) classified cedar boards into five quality classes with 90% accuracy. For a comprehensive overview of wood defect detection research, consult (Kryl et al., 2020). The work by Norell (2009) introduces an automated technique for quantifying annual rings in noisy sawmill images of logs, enabling the identification of a single quality attribute. In comparison to an X-ray scanner at approximately 85% accuracy (Niemz et al., 2023), our system's performance is only slightly lower. However, it is crucial to acknowledge that our dataset lacks the full complexity.

Table 3 shows an overview of the species model results. The best model is EfficientNetB0 with an accuracy of 91% and an F1 score of 0.88. The confusion matrix is shown in Fig. 13. Fig. 14 shows the accuracy (a) and loss (b) plot for the training and validation dataset of EfficientNetB0. ReNet50 reaches an accuracy of 90% with an F1 score of 0.87, followed by VGG19 and InceptionNetV3. The confusion matrix is shown in Fig. 12. All other accuracy, loss plots, and confusion matrices of quality and species models can be found in Appendix C.

The models were also tested with simple augmentation, like flipping and rotating, but this made the results slightly worse (0.01%). This can be due to the already existing high variance in image brightness,

Table 3
Results of species models.

Model	Avg. val acc/loss	Avg. val F1 score	Test acc/loss	Test F1 score
ResNet50	0.92 (±0.01)/ 0.41 (±0.05)	0.89 (±0.01)	0.90/0.47	0.87
EfficientNetB0	0.91 (±0.01)/ 0.37 (±0.03)	0.90 (±0.01)	0.91/0.37	0.88
VGG19	0.90 (±0.01)/ 0.71 (±0.04)	0.87 (±0.01)	0.88/0.78	0.85
InceptionNetV3	0.84 (±0.04)/ 0.86 (±0.19)	0.80 (±0.03)	0.83/0.86	0.80

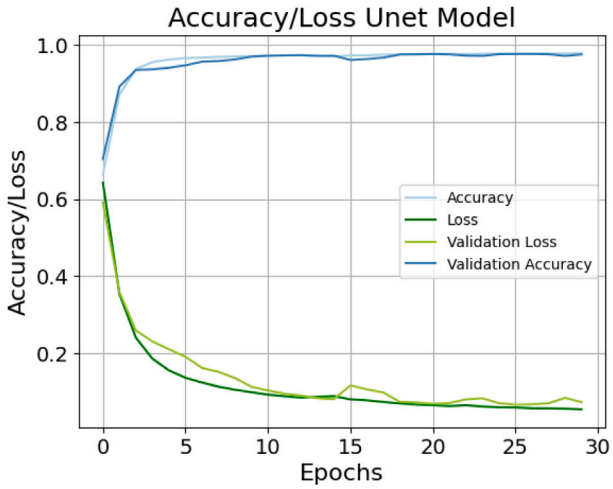


Fig. 8. Accuracy and loss over 30 epochs for training and validation of the Unet segmentation model.

shapes, and contrast. Nevertheless, in the future, it might be useful to apply augmentation techniques to make models more robust.

The literature primarily deals with species classification on wooden boards using magnified images. One example is the CAIRO dataset (Kryl et al., 2020), which magnifies images by 10x and covers over 100 tropical species. Most research focuses on tropical wood species, with models achieving accuracies ranging from 72% to 99.84% (Kryl et al., 2020). In another study, Fabijańska, Danek, and Barniak (2021) achieved 98.7% accuracy in distinguishing 14 European tree species using wood core images. Compared to prior studies, our dataset is being acquired in a real-time process rather than a controlled experimental environment, which is generally more difficult. Despite this we can still achieve a respectable 91% accuracy.

We conducted an in-depth analysis to explore the potential benefits of enhancing either the resolution or the quantity of training samples. Fig. 15 shows a positive correlation between resolution and accuracy/F1 score for ResNet50 quality and species model. It shows asymptotic behavior and the increase in accuracy is merely 1% between 512 and 1024 pixels. This indicates that the used resolution of 1024 pixels is sufficient.

Fig. 16 shows the impact of the number of samples on the accuracy/F1 score for quality (a) and species (b) model built upon the ResNet50 architecture. Accuracy and F1 score tend to increase as the dataset's size expands. Notably, the quality prediction model demonstrates a considerably more pronounced increase than the species classification. This leads to the hypothesis that the quality model stands to experience substantial enhancements through the introduction of a larger and more diverse training dataset, while the species model can also be improved, but not significantly.

It is also interesting how long it takes to train the models and make predictions for individual samples. Plot Fig. 17 shows an overview of the training times for the quality and species models (5-fold cross-validation and final model each). It is important to note that the models are trained for a different number of epochs to achieve their best accuracy. The number of epochs is denoted in blue. For the quality model, EfficientNetB0 final model takes the longest time, followed by ResNet50, InceptionNetV3, and VGG19. For the species model, ResNet50 needs the most time, followed by InceptioNetV3, EfficientNetB0, and VGG19. ResNet50 has the fastest runtime per epoch (4.3 min for the species model, 1 min for the quality model), followed by EfficientNet (5.5 min for the species model, 1.2 min for the quality model). Due to the higher number of data, the species models need a significantly longer time to train than the quality models.

We further analyzed the prediction times for single images. Table 4 shows an overview of the prediction times for each quality and species model as well as the time needed to preprocess one image. Therefore three different images were randomly selected, preprocessed, and put through each quality and species model. The average time needed for preprocessing (including the segmentation) was 16.1 s. The fastest model for prediction is VGG19 with 24.7 s overall, followed by ResNet50 and EfficientNetB0, while InceptionNetV3 is the slowest. For the time predictions, we used a sequential computing approach as described in Section 4.6. By parallelizing the process, the overall prediction time can be accelerated. This includes for example computing the species and the quality model at the same time.

To find a suitable model for deployment in the process, both prediction time and the model's accuracy play a role. Depending on the requirements, these considerations may vary from one sawmill to another. However, it is generally important to fulfill both requirements as effectively as possible, which is why we recommend ResNet50. Compared to EfficientNetB0, which exhibits the highest accuracy, ResNet's accuracy is merely 1% lower for species and equivalent for the quality model, showcasing notably faster prediction times.

To evaluate the models, the throughput, i.e. the number of evaluations per minute, was considered for Resnet50 using a sequential computing approach. The throughput of a single GPU is 1.9stems/min. By harnessing the computational power of 3 GPUs/ calculating 3 predictions in parallel, the throughput is 5.7stems/min and it already surpasses the existing throughput of 4 stems per minute within the current operational process of our partner sawmill. However, it must be noted that an installation of the camera after the cut (position 4) would reduce the throughput. The time that the trunk is in the process before the image is taken must be added. This underscores a pivotal rationale for advocating the camera's placement as expounded in this paper.

6. Conclusion

This paper provides a proof-of-concept for sorting roundwood based on quality and species. It shows a successful image acquisition as well as models that are able to learn quality and species features from a selected dataset. The system offers a flexible, adaptable, and scaleable solution for automating the process. The same software can be used for several process lines and only the costs of the camera and installation are incurred. With the help of different methods such as

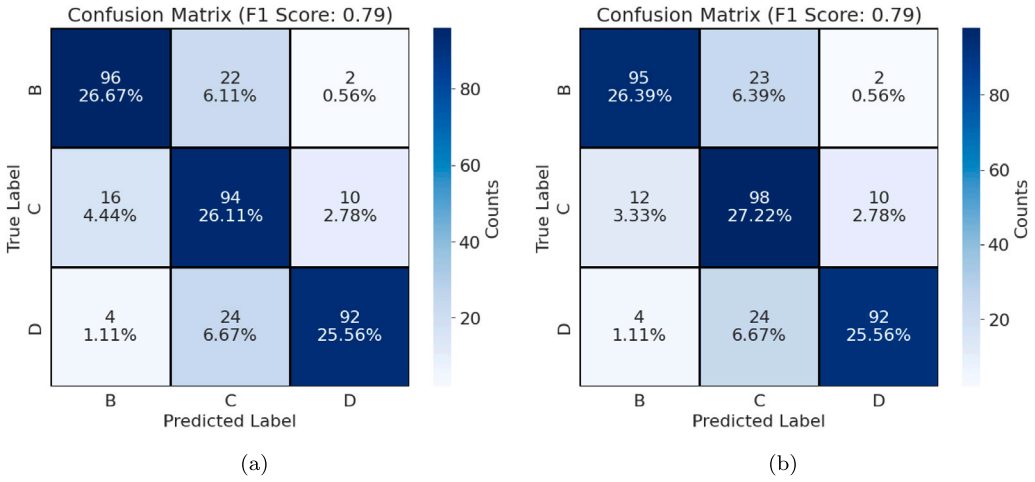


Fig. 9. Confusion matrices of the final quality model of ResNet50. (a) without the numerical constraint, (b) with numerical constraint for taper.

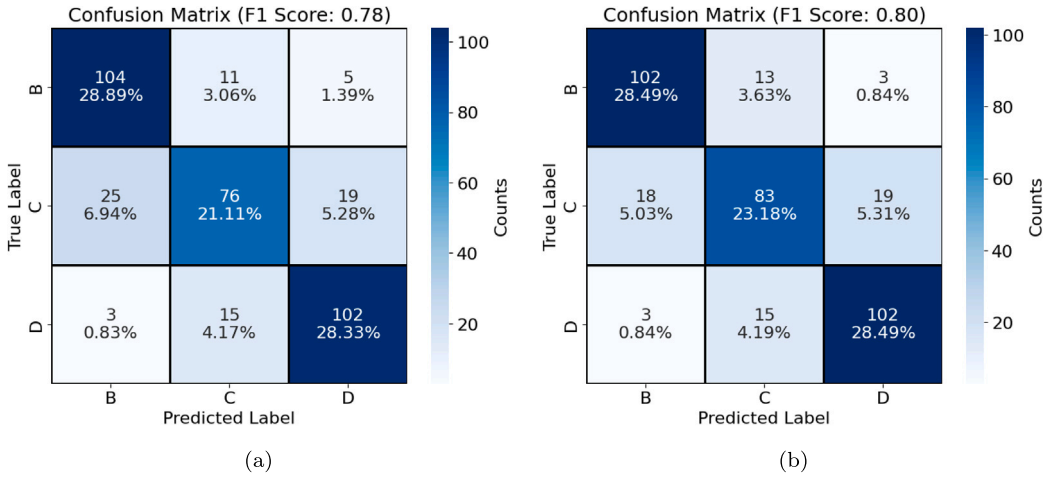


Fig. 10. Confusion matrices of the final quality model of EfficientNetB0. (a) without numerical constraint, (b) with numerical constraint for taper.

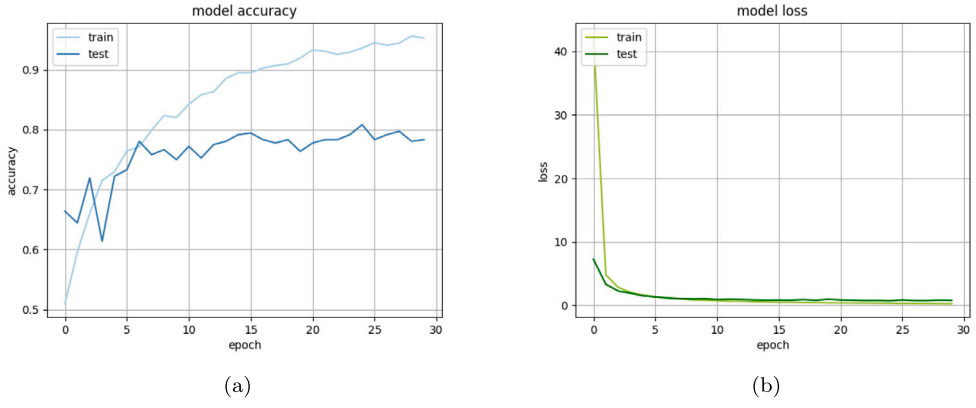


Fig. 11. (a) Accuracy plot and (b) loss plot of final ResNet50 model over 30 epochs.

Table 4
Prediction times for perprocessing, quality and species model for single images with sequential computing approach.

Model	Time for preprocessing	Time for quality model	Time for species model	Overall time
ResNet50	16.1 s	7.6 s	7.7 s	31.4 s
EfficientNetB0	16.1 s	12.8 s	11.0 s	39.9 s
VGG19	16.1 s	4.3 s	4.3 s	24.7 s
InceptionNetV3	16.1 s	12.3 s	13 s	41.4 s

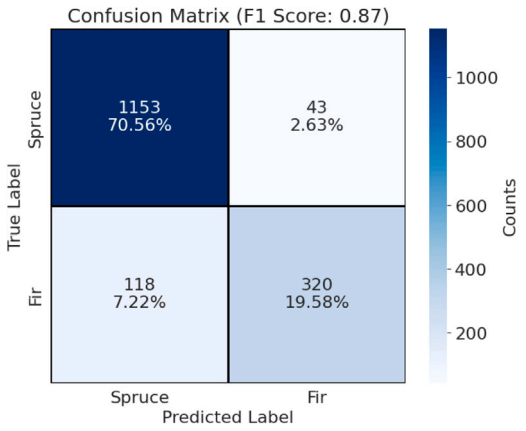


Fig. 12. Confusion matrix of the final species model of ResNet50.

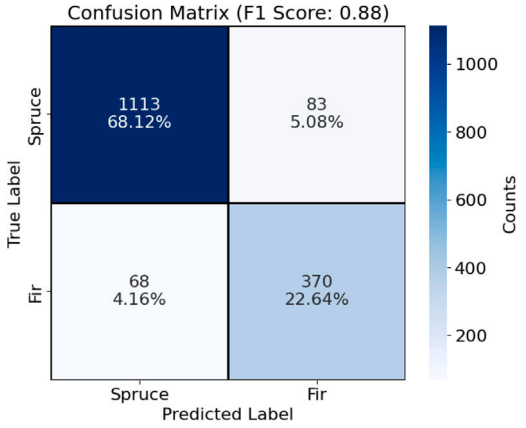


Fig. 13. Confusion matrix of the species quality model of EfficientNetB0.

transfer learning, the adaptability to domain shifts can be implemented. For example, for the use in a different sawmill or the occurrence of other species.

However, it is important to note that the used quality dataset does not reflect real-world complexity. The training and prediction with the overall quality dataset remains a challenge. Future research is planned to overcome this problem and train more robust models. Therefore a second image of the trunk is necessary to reduce the missing information in the dataset. Asynchronized data should be reduced as much as possible manually and with the help of hardware and software. In order to deal with the remaining noise, approaches such as loss corrections should be applied. To further improve the accuracy, special architectures for fine-grained image analysis and ensemble models should be used.

Looking ahead, the industrial deployment of the system entails several objectives. Enhancing model explainability and analyzing the uncertainty of predictions are paramount for engendering trust and providing valuable intervention insights. As data accumulates, the pursuit of encompassing the four unexplored quality classes and additional species in model learning is anticipated. In order to improve throughput and reduce the number of GPUs and computing power required, a common model for quality and species could be trained. Moreover, it is necessary to optimize the code in terms of time and effort and build the necessary infrastructure. This also includes sufficient version control management, code reviews and continuously updating the models (Paleyes, Urma, & Lawrence, 2022).

In summation, this paper validates the feasibility of automating log sorting. However, for seamless integration into real-world operations, specific enhancements are imperative. This holistic perspective underscores both the achievements and the ongoing strides necessary to actualize the potential of the proposed automation framework.

CRediT authorship contribution statement

Julia Achatz: Conceptualization, Methodology, Software, Investigation, Data curation, Visualization, Writing – original draft. **Mirko Lukovic:** Review & editing. **Simon Hilt:** Software, Validation, Methodology, Investigation. **Thomas Lädrach:** Resources, Data curation, Writing – review & editing. **Mark Schubert:** Supervision, Project administration, Funding acquisition, Conceptualization, Writing – review & editing.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Mark Schubert reports financial support was provided by Innosuisse - Swiss Innovation Agency. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work the authors used ChatGPT in order to improve the written text. After using this tool/service, the authors reviewed and edited the content as needed and takes full responsibility for the content of the publication.

Acknowledgments

We express our gratitude to Mr. Erich Salzmann from OLWO AG for providing us with the quality dataset and lending his expertise in roundwood inspection. Financial support for this project was provided by Innosuiss through funding grant [51566.1 IP-ENG].

Appendix A. Detailed quality features

This section provides detailed quality feature calculations on the cross-section and the stem. Fig. A.18 shows the most important features on the cross section including the formulas for calculating their expressions. The table in Fig. A.19 shows the feature on the stem.

Appendix B. Hyperparameter

The following hyperparameters are fixed for all final models:

- Resolution: 1024 x 1024 pixel
- Color channels: 3
- Batch size: 4
- Balancing strategy: reweighting for species, None for quality
- Augmentation: No augmentation
- Transfer learning: based on imagenet
- Stratified 5 fold cross validation
- Optimizer: Adam with learning rate of 0.00001 for quality and 0.000001 for species, $beta_1 = 0.8$ and $beta_2 = 0.9$
- Early stopping: max epochs: 300 patience of 5 based on val loss

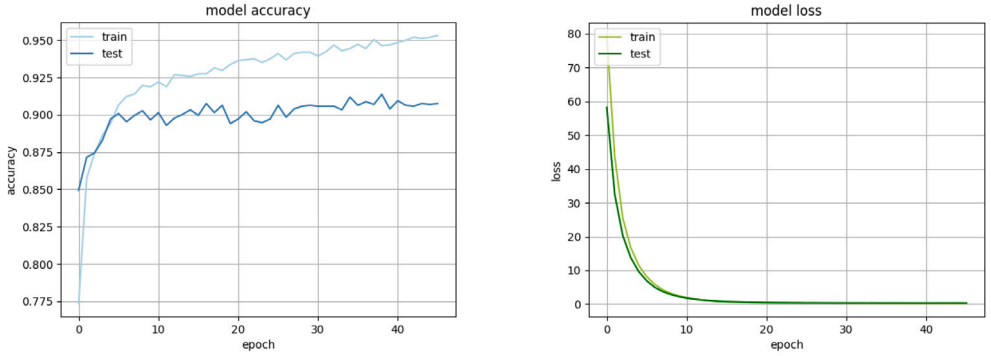


Fig. 14. (a) Accuracy plot and (b) loss plot of final EfficientNetB0 species model over 46 epochs.

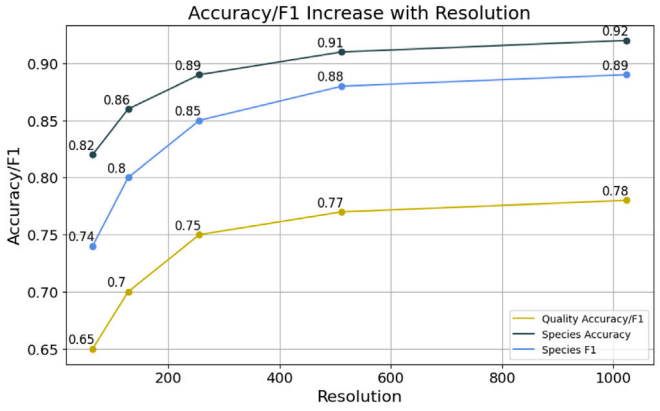


Fig. 15. Accuracy/F1 score with increasing resolution for species and quality ResNet50 model.

- No cleaning of overexposed/blurry images (left to keep models robust)

Manually tuned hyperparameters:

- ResNet50 quality: $dr = 0.6$, alpha $L2 = 0.4$, $alphaL2_2 = 0.01$, number of frozen layers: 165
- EfficientNetB0 quality: $dr = 0.6$, $alphaL2 = 0.4$, $alphaL2_2 = 0.01$, number of frozen layers: 100
- VGG19 quality: $dr = 0.7$, $alphaL2 = 0.5$, $alphaL2_2 = 0.05$, number of frozen layers: 10
- InceptionNetV3 quality: $dr = 0.6$, $alphaL2 = 0.4$, $alphaL2_2 = 0.01$, number of frozen layers: 100
- ResNet50 species: $dr = 0.7$, $alphaL2 = 0.6$, $alphaL2_2 = 0.01$, number of frozen layers: 165
- EfficientNetB0 species: $dr = 0.6$, $alphaL2 = 0.4$, $alphaL2_2 = 0.01$, number of frozen layers: 100
- VGG19 species: $dr = 0.6$, $alphaL2 = 0.4$, $alphaL2_2 = 0.1$, number of frozen layers: 10
- InceptionNetV3 species: $dr = 0.7$, $alphaL2 = 0.5$, $alphaL2_2 = 0.1$, number of frozen layers: 100

These are the number of parameter each model has:

- ResNet50 quality: Total parameters: 90.697.091; Trainable parameters: 71.575.043; Non-trainable parameters: 19.122.048
- EfficientNetB0 quality: Total parameters: 45.993.126; Trainable parameters: 45.783.903; Non-trainable parameters: 209.223
- VGG19 quality: Total parameters: 36.802.115; Trainable parameters: 35.066.627; Non-trainable parameters: 1.735.488
- InceptionNetV3 quality: Total parameters: 80.785.699; Trainable parameters: 78.609.283; Non-trainable parameters: 2.176.416
- ResNet50 species: Total parameters: 90.696.962; Trainable parameters: 71.574.914; Non-trainable parameters: 19.122.048
- EfficientNetB0 species: Total parameters: 45.992.997; Trainable parameters: 45.783.774; Non-trainable parameters: 209.223
- VGG19 species: Total parameters: 36.801.986; Trainable parameters: 35.066.498; Non-trainable parameters: 1.735.488
- InceptionNetV3 species: Total parameters: 80.785.570; Trainable parameters: 78.609.154; Non-trainable parameters: 2.176.416

Appendix C. Detailed results

This section shows the confusion matrices, loss and accuracy plots of ResNet50, EfficientNetB0, VGG19 and InceptionNetV3 for quality and species, not shown in the main text (see Figs. C.20–C.28).

Appendix D. Label noise calculation

D.1. Label noise by synchronization

A synchronization error happened on 38% of the 16 observed days. With the assumption that the mismatch happens after 50% of the time, 19% of all samples are affected by the synchronization problem. In order to answer the question of how many labels are wrong, we calculate the probability of an affected sample to get the right label by chance:

- Probability of an instance to belong to class B: $P(B) = 0.43$
- Probability of an instance to belong to class C: $P(C) = 0.5$
- Probability of an instance to belong to class D: $P(D) = 0.07$
- Probability that a B instance gets a B label reassigned: $P(B) * P(B) = 0.1849$
- Probability that a C instance gets a C label reassigned: $P(C) * P(C) = 0.25$
- Probability that a D instance gets a D label reassigned: $P(D) * P(D) = 0.0049$
- Probability that one instance gets the right label: $0.1849 + 0.25 + 0.0049 = 0.4398$
- Probability that the technical mismatch problem causes an actual change of labels: $(1 - 0.4398) * 0.19 = 0.106$

D.2. Estimation of label noise

Since one sample can be affected by different kinds of label noise at the same time, we need to calculated the probability that a label is wrong compared to the ground truth. For that we are looking at the worst case and a best case scenario. In the worst case the occlusion

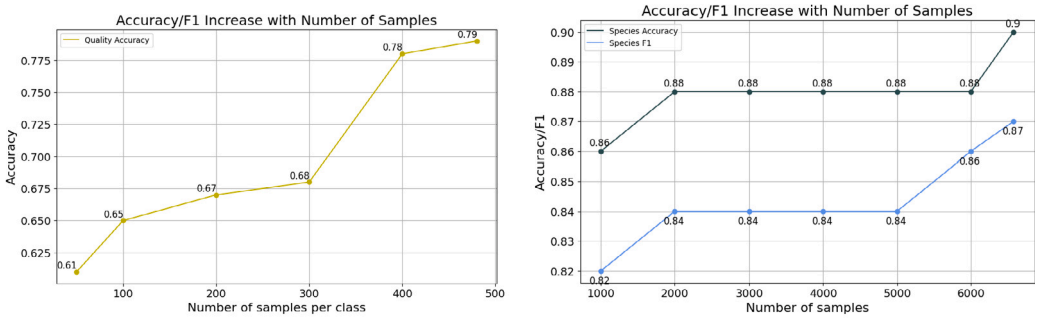


Fig. 16. Accuracy/F1 score increase with increasing number of samples per class for (a) quality model and (b) species model.

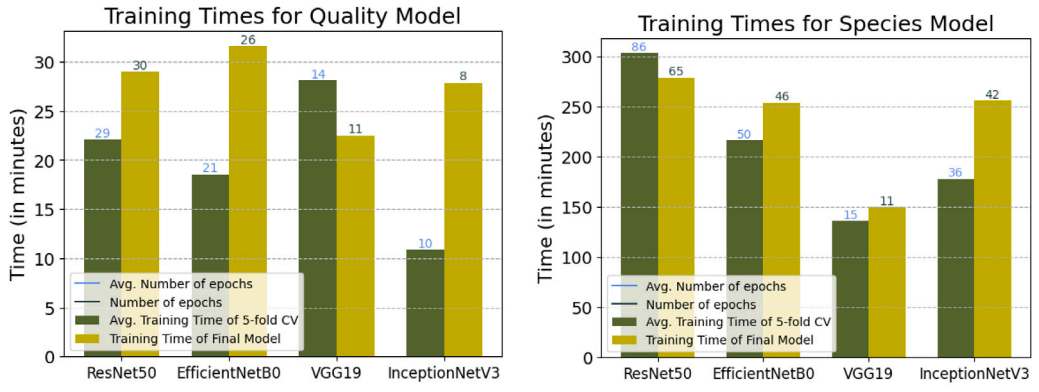


Fig. 17. Training times for (a) quality model and (b) species model.

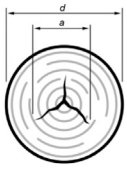
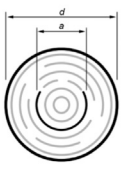
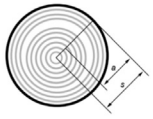
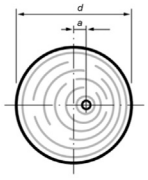
Feature	Cracks	Ring dish	Annual Rings	Reaction Wood	Ecentric growth
Visualization					
Formula	$\frac{a}{d} * 100 = x\ \%$	$\frac{a}{d} * 100 = x\ \%$	$\frac{a}{n} * 10 = x\ mm$ $a = 0.75 * s$ N: Number of rings within a	$\frac{a}{d} * 100 = x\ \%$	$\frac{a}{d} * 100 = x\ \%$

Fig. A.18. Features which can be seen on the cross section of a stem, which are important for quality classification including formulas. Based on Bundesamt für Umwelt BAFU (2010).

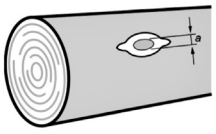
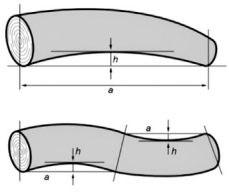
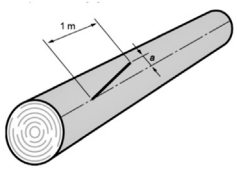
Feature	Branches	Curvature	Spiral Growth
Visualization			
Formula	a: diameter in cm	$\frac{h}{a} = x\ cm/m$	a: Fiber inclination in cm

Fig. A.19. Features which can be seen on the stem, which are important for quality classification including formulas. Based on Bundesamt für Umwelt BAFU (2010).

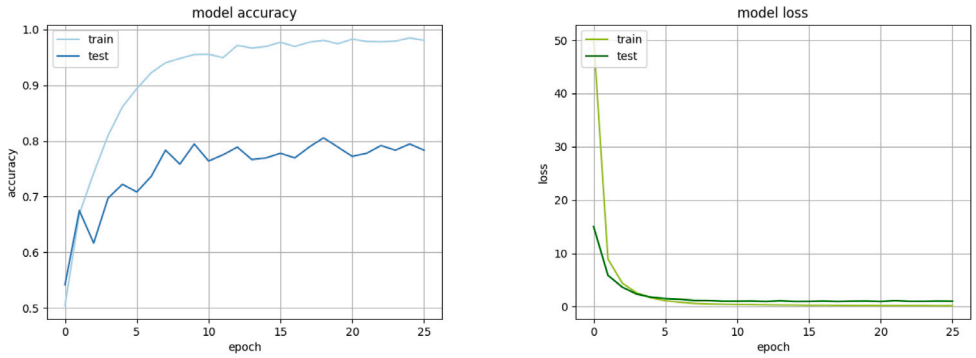


Fig. C.20. (a) Loss plot and (b) accuracy plot of final quality EfficientNetB0.

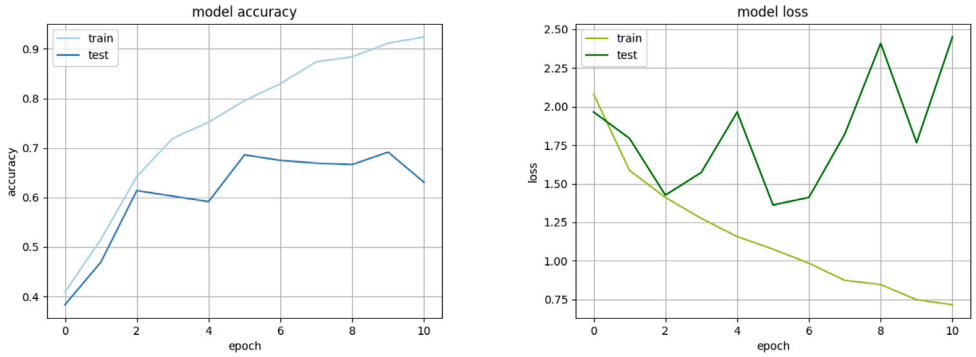


Fig. C.21. (a) Loss plot and (b) accuracy plot of final quality VGG19. Intense overfitting results in a rise in test loss, a challenge that can be mitigated by expanding the dataset in subsequent iterations.

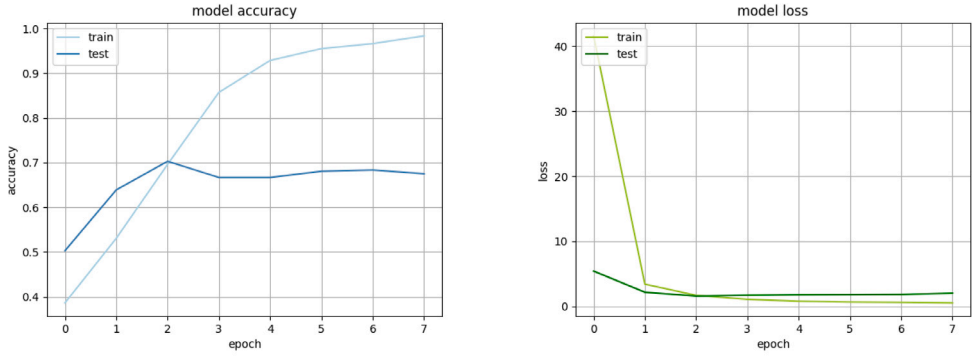


Fig. C.22. (a) Loss plot and (b) accuracy plot of final quality InceptionNetV3.

noise has the probability P(O) = 0.34 and the missing information has P(I) = 0.25. The human error is P(H) = 0.15 and the previously technical mismatch is at P(T) = 0.106. The probability that a label is wrong in the dataset can be approximates with the following calculation:

$$P(O)+P(I)+P(H)+P(T)-P(O,I)-P(O,H)-P(O,T)-P(I,H)-P(I,T)-P(H,T)+P(O,I,H)+P(O,I,T)+P(I,H,T)+P(H,T,O)-P(O,I,H,T)=0.34+0.25+0.15+0.106-0.085-0.051-0.03604-0.0375-0.0265-0.0159+0.01275+0.00901+0.003975+0.005406-0.0013515=0.62$$

Looking at the best case in which only the human error P(H) = 0.15 and the technical mismatch exists P(T) = 0.106. The resulting probability of having label noise is at: $P(H)+P(T)-P(H,T)=0.256-0.0159=0.2401$ In conclusion the label noise lies in between 24% and 62%.

D.3. Results of noisy and blur detector

The results of the image noise detectors are shown in Fig. D.29. The blur detector (a) reaches an F1 score of 0.84. The overexposed detector (b) reaches an F1 score of 0.87.

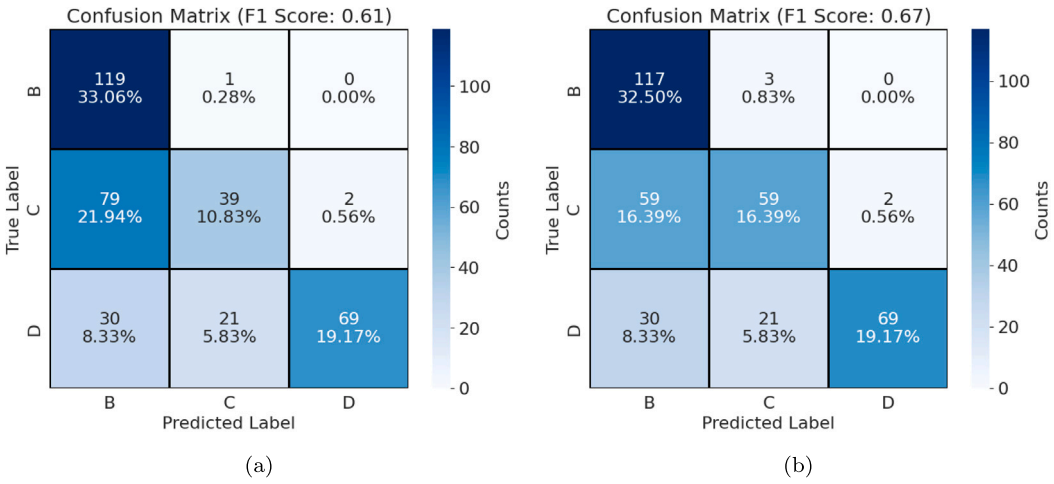


Fig. C.23. Confusion matrices of VGG19 quality model (a) without and (b) with numerical constraint.

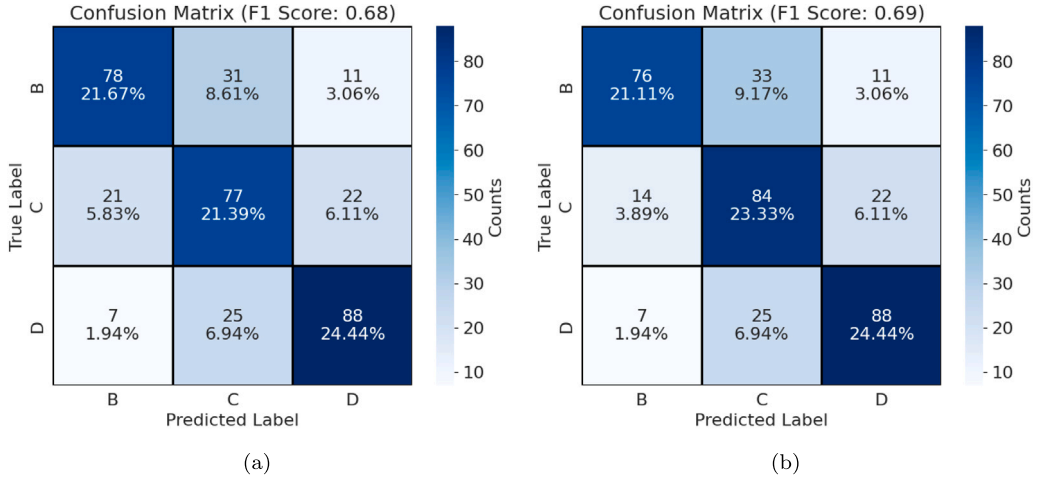


Fig. C.24. Confusion matrices of InceptionNetV3 quality model (a) without and (b) with numerical constraint.

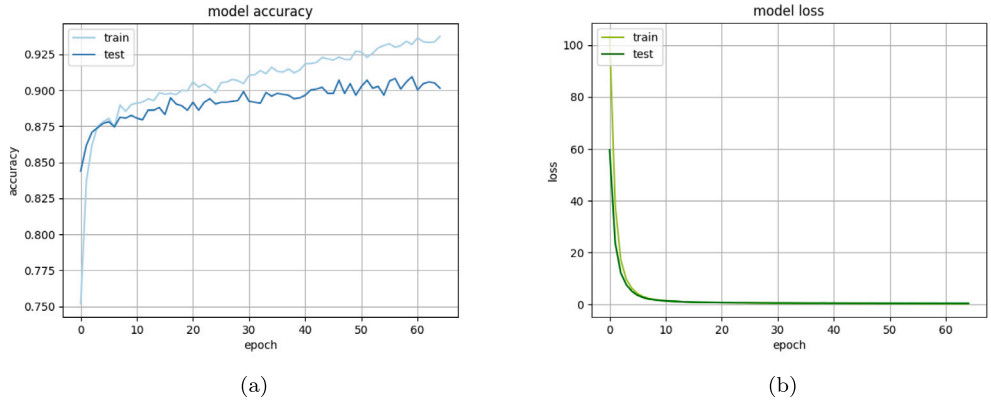


Fig. C.25. (a) Loss plot and (b) accuracy plot of final species ResNet50.

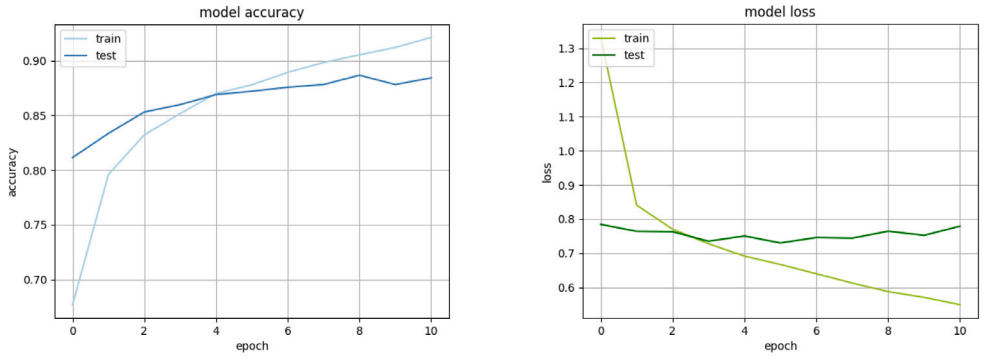


Fig. C.26. (a) Loss plot and (b) accuracy plot of final species VGG19.

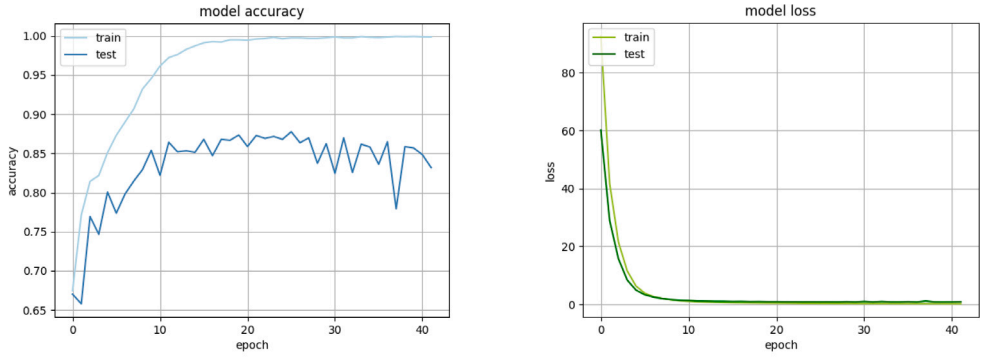


Fig. C.27. (a) Loss plot and (b) accuracy plot of final species InceptionNetV3.

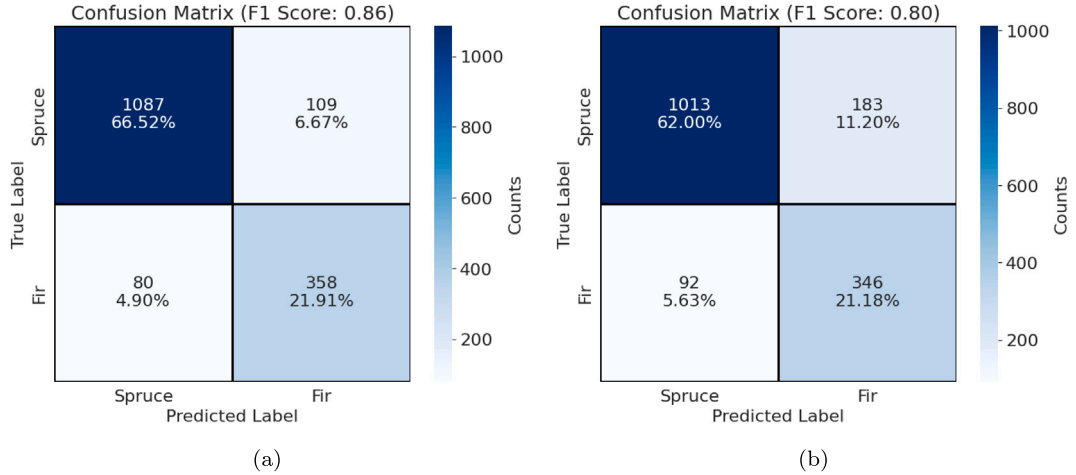


Fig. C.28. Confusion matrices of (a) VGG19 model and (b) InceptionNetV3.

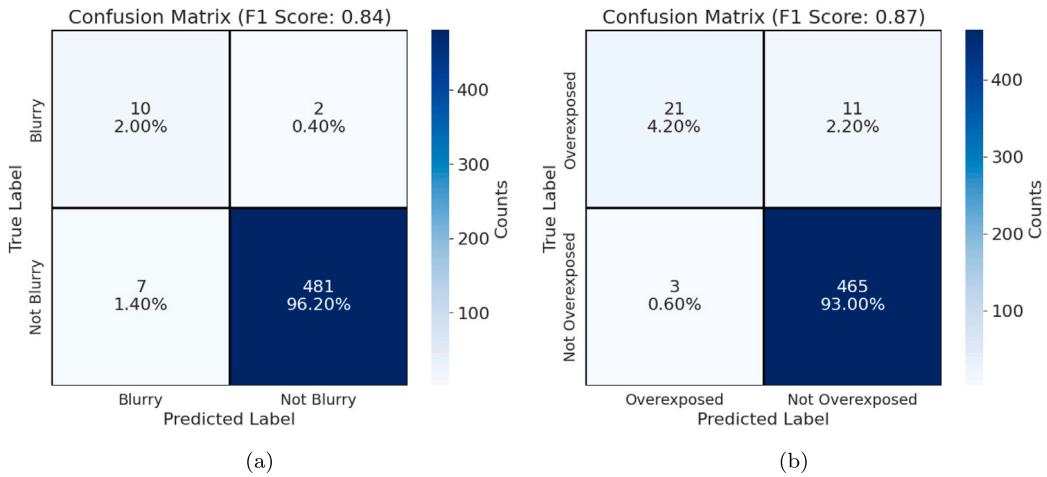


Fig. D.29. Confusion matrices of Noise Detectors. (a) Blur detector tested on 500 samples, (b) Overexposed detector tested on 500 samples.

References

Affonso, C., Rossi, A. L. D., Vieira, F. H. A., de Leon Ferreira, A. C. P., et al. (2017). Deep learning for biological image classification. *Expert Systems with Applications*, *85*, 114–122.

Björklund, A., Husfeldt, T., & Koivisto, M. (2009). Set partitioning via inclusion-exclusion. *SIAM Journal on Computing*, *39*(2), 546–563.

Bundesamt für Umwelt BAFU (2010). Schweizer handelsgebräuche für rohholz.

Chen, L., Li, S., Bai, Q., Yang, J., Jiang, S., & Miao, Y. (2021). Review of image classification algorithms based on convolutional neural networks. *Remote Sensing*, *13*(22), 4712.

Fabijańska, A., Danek, M., & Barniak, J. (2021). Wood species automatic identification from wood core images with a residual convolutional neural network. *Computers and Electronics in Agriculture*, *181*, Article 105941.

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., et al. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.

Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4700–4708).

Kaiming, H., Xiangyu, Z., Shaoqing, R., & Jian, S. (2015). Deep residual learning for image recognition. CoRR abs/1512.03385.

Kotsiantis, S., Kanellopoulos, D., Pintelas, P., et al. (2006). Handling imbalanced datasets: A review. *GESTS International Transactions on Computer Science and Engineering*, *30*(1), 25–36.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In F. Pereira, C. Burges, L. Bottou, & K. Weinberger (Eds.), *Advances in neural information processing systems, Vol. 25*. Curran Associates, Inc..

Kryl, M., Danys, L., Jaros, R., Martinek, R., Kodytek, P., & Bilik, P. (2020). Wood recognition and quality imaging inspection systems. *Journal of Sensors*, *2020*, 1–19.

Liu, F., & Panagiotakos, D. (2022). Real-world data: a brief review of the methods, applications, challenges and opportunities. *BMC Medical Research Methodology*, *22*(1), 287.

Niemz, P., Teischinger, A., & Sandberg, D. (2023). *Springer handbook of wood science and technology*. Springer.

Norell, K. (2009). An automatic method for counting annual rings in noisy sawmill images. In *Image analysis and processing–ICIAP 2009: 15th international conference vietri sul mare, Italy, september 8-11, 2009 proceedings 15* (pp. 307–316). Springer.

Nurthohari, Z., Murti, M. A., & Setianingsih, C. (2019). Wood quality classification based on texture and fiber pattern recognition using hog feature and svm classifier. In *2019 IEEE international conference on internet of things and intelligence system (IoTaIS)* (pp. 123–128). IEEE.

O'Mahony, N., Campbell, S., Carvalho, A., Harapanahalli, S., Hernandez, G. V., Krpalkova, L., et al. (2020). Deep learning vs. traditional computer vision. In *Advances in computer vision: Proceedings of the 2019 computer vision conference (CVC), volume 1 1* (pp. 128–144). Springer.

Paleyes, A., Urma, R.-G., & Lawrence, N. D. (2022). Challenges in deploying machine learning: a survey of case studies. *ACM Computing Surveys*, *55*(6), 1–29.

Rokach, L. (2009). Taxonomy for characterizing ensemble methods in classification tasks: A review and annotated bibliography. *Computational Statistics & Data Analysis*, *53*(12), 4046–4072.

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18* (pp. 234–241). Springer.

Rosenbrock, A. (2015). Blur detection with opencv. Retrieved from https://pyimagesearch.com/2015/09/07/blur-detection-with-opencv Accessed September 01, 2023.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

Song, H., Kim, M., Park, D., Shin, Y., & Lee, J.-G. (2022). Learning from noisy labels with deep neural networks: A survey. *IEEE Transactions on Neural Networks and Learning Systems*.

Song, T., Li, H., Meng, F., Wu, Q., & Cai, J. (2017). LETRIST: Locally encoded transform feature histogram for rotation-invariant texture classification. *IEEE Transactions on Circuits and Systems for Video Technology*, *28*(7), 1565–1579.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S. E., Anguelov, D., et al. (2014). Going deeper with convolutions. CoRR abs/1409.4842.

Tan, M., & Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning* (pp. 6105–6114). PMLR.

Wang, J., Ma, Y., Zhang, L., Gao, R. X., & Wu, D. (2018). Deep learning for smart manufacturing: Methods and applications. *Journal of Manufacturing Systems*, *48*, 144–156.

Wang, H., Wang, H., Yu, W., & Li, H. (2019). Research on wood species recognition method based on hyperspectral image texture features. In *2019 4th international conference on mechanical, control and computer engineering (ICMCCE)* (pp. 413–4133). IEEE.

Xu, J., Kovatsch, M., Mattern, D., Mazza, F., Harasic, M., Paschke, A., et al. (2022). A review on ai for smart manufacturing: Deep learning challenges and solutions. *Applied Sciences*, *12*(16), 8239.