Integrating Attention Mechanism and Optimizers for Enhanced Wood Surface Analysis and classification with a Novel Dataset

Rohini A. Bhusnurmath and Shaila Doddamani *
Department of Computer Science,
Karnataka State Akkamahadevi Women University,
Vijayapura-586108, Karnataka, India

Email:rohiniabmath@gmail.com *Email:dodamanishaila@gmail.com*

ABSTRACT:

The wood industry faces significant challenges due substantial variability in raw materials and the complexity of manufacturing processes, which result in numerous visible structural defects. Manual quality control, reliant on trained specialists, is often tedious, biased, and less effective. Automated vision-based systems have been proposed as a solution, achieving higher recognition rates than human experts. However, the field suffers from a lack of large-scale, authentic datasets that encompass both normal and defective wood surface images.
To address this, a comprehensive dataset is created by merging two distinct sets: one with images of normal wood surfaces and another with images of defective wood surfaces. Using this dataset, authors developed and evaluated deep learning models for wood surface defect classification, using ResNet18 and ResNet18 with a bottleneck attention mechanism.
Experimental results show that ResNet18 achieved an accuracy of 77.14%, while the enhanced model with bottleneck attention significantly improved performance, reaching an accuracy of 94.86%. Also compared various optimization algorithms to determine their impact on model performance, finding that the choice of optimizer crucially affects overall accuracy.
The proposed study provides a valuable dataset and demonstrates that integrating attention mechanisms into deep learning models, along with careful selection of optimizers, can significantly enhance the performance of automated vision-based quality control systems in the wood industry.

Keywords: Deep Learning, ResNet18, Bottleneck Attention Mechanism, Wood Surface Analysis, Texture classification.

1. INTRODUCTION:

Wood is a very valuable commodity and finds its application in many industries for aesthetic and structural purposes, as well as its value. Imperfections in the product can considerably lower the marketable value of the wood. Veneers are one of the most common forms of wood used industrially in making furniture and for decorative purposes; they are very prone to defects like live knots, dead knots, and cracks. These defects could either be due to low-quality raw materials or to poor manufacturing methods [1, 3]. In many developing countries, these defects lead to a substantial underutilization of raw timber resources, further compounding economic and material waste. Traditionally, the wood processing industry relies heavily on skilled personnel to conduct visual quality inspections, a method that is both time-consuming and subjective [2, 16]. To optimize wood utilization rates and boost profitability, there is a pressing need for quick and accurate defect detection methods in the wood processing industry. Various advanced techniques have been explored to address this need. Among them methods, computer vision has gained significant attention due to its ability to automate the inspection process and its potential for integration into intelligent processing systems. Computer vision technology can detect and classify various wood defects such as cracks, knots, and marrows, preserving the wood's strength and texture by ensuring only high-quality products proceed to market.

Despite these advancements, traditional solid-wood panel detection has predominantly relied on manual identification, processing, and marking of defects. This approach is not only labor-intensive but also prone to human error. Since the turn of the twenty-first century, the wood industry has been focusing on researching and developing intelligent processing technologies [19]. Image processing, one of the most recent advancements, has seen extensive application in wood defect detection.

In this study, the objective is to augment wood defect classification through the utilization of deep learning methodologies. The primary focus lies in the precise identification and categorization of normal and distinct defect types. By employing advanced deep learning techniques, the aim is to furnish a resilient solution for the automated and precise categorization of wood defects. Following are the contributions of the proposed study:

- Aim is to merge the datasets of normal wood and defect classes for comprehensive real-world representation.
- Implementing ResNet18 as a baseline, to check accuracy and dataset complexity.
- To identify bias risk in merged datasets, affecting model generalizability.
- To demonstrate bottleneck attention mechanism in ResNet18 for compressing and standardizing essential features.
- To check bottleneck mechanism’s role in reducing dimensionality and focusing on critical features.
- To find accuracy improvement with bottleneck- attention mechanism ResNet18, effectively distinguishing between normal and defect types.

This paper is organized as follows: the next section deals with associated related work about wood surface defect classification; Section 3 proposes methods in detail; further, Section 4 emphasizes the results of experiments and discussion; and finally, Section 5 gives the conclusion and discusses the future work.

2. LITRATURE SURVEY

Aggarwal and Kumar [4] have worked on surface texture classification using deep learning models to reduce computational costs associated with large training datasets. They proposed a convolutional neural network model, and that is divided into two sub-models, designed with customized parameters to classify textures efficiently using fewer samples. Utilized the Kylberg Texture dataset with 16 texture classes, their models achieved accuracies of 92.42% for model-1 and 96.36% for model-2. These models outperform conventional techniques, balancing accuracy and computational efficiency.

Zhouxin et al. [5] highlighted the need for identification of tree species in natural forests based on the fine-scale traits extracted from TLS point clouds. Thirteen machine learning and deep learning classifiers have been used for species classification, and their filtering wood points have been compared based on characteristics of trees by 15 classifiers based on mean Intersection over Union accuracy (mIoU), training stability, and/or time cost. Wood classification and species classification mIoU achieved 10% and 5% better with deep learning methods than machine learning ones. For the highest species classification mIoU of 0.906, it was achieved with PointNet++. For wood classification, the best mIoU score was 0.839 with UNet. We further analyzed the resolution of the input, attributes, and features of classification. Shihui et al. [6] tackled the requirement of classification of wood type with very high accuracy for construction and furniture purposes. They proposed an SSR-based CNN structure that is automatically capable of learning features from wood images. The low-level challenges of the traditional methods based on handcrafting are taken care of. The SSR module combines channel split and shuffle operations into the residual structure for minimizing computational costs with no accuracy degradation. Their model achieved 94.86% accuracy, with processing taking 26.55 ms to complete a single image, thus outperforming all traditional methods and other deep learning networks; it will therefore be suitable for real-time wood classification.
Deivison et al. [7] developed an advanced species recognition system to aid in flora conservation, using 1901 wood images from 46 Brazilian species. They compared early fusion of Local Binary Patterns and late fusion at the decision classifier stage. The system, employing an SVM classifier with rotation-invariant LBP histograms, achieved a 97.67% F1-score, improved slightly by majority voting. It effectively differentiated species, including those often misclassified by experts, showing high accuracy and potential for reducing identification errors when combined with traditional methods.

Anna et al. [8] developed a convolutional neural network with residual connections approaches for automatic identification of tree species from scanned wood core images. Using sliding window strategy and majority voting, the model achieved 93% accuracy for patch classification and 98.7% for core images, outperforming a state-of-the-art competitor. The study analyzed the influence of model parameters and made the source code and dataset available for replication.
Joongbin et al. [9] developed a model that classifies five dominant tree species in North Korea by using satellite data and machine-learning techniques. When tested in the Gwangneung Forest in South Korea, Mt. Baekdu in China, and Goseong-gun in the North-South Korea border, this model achieved accuracies of 83%, 91%, and 90%, respectively. A combined model was developed by training the models with data from all three areas combined. It achieved 80% accuracy at Goseong-gun, South Korea. Then, this model was used to predict the dominant tree species in Goseong-gun, North Korea, showing its regional applicability and thereby demonstrating its potential for mapping areas that are inaccessible. Mohd et al. [10] applied unmanned aerial vehicles for the classification of forest area classes by leveraging deep

Learning-based supervised image classification algorithms.

They resorted to stacked autoencoders and demonstrated its efficiency in the accurate estimation of forest cover. The experimental results testified to deep learning performing much better compared to the various traditional machine learning algorithms, with an overall accuracy of 93% by cross-validation. Their work underlines the application relevance of UAV-based RS and deep learning in forestry, particularly with respect to monitoring deforestation and encroachment threats. Yago et al. [11] alluded to the importance of forests as a filter for CO2, as well as its socio-economic importance. They underscored the role of drones, these days, in collecting forest data cost-effectively. They reviewed studies on using RGB images from UAVs for deep learning for various problems in forestry research, including tree detection, species classification, and anomaly detection, such as fire in forests. This review will therefore address the strengths, challenges in the methodology, and the possible resources available in the domain for the research fraternity.

Fenglong et al. [12] addressed the problem of wood defect detection using machine vision and deep learning methods, such approaches being extremely necessary for labor-intensive and inefficient manual inspection processes. Surface images were taken with a color charge-coupled device camera from Akagi and Pinus sylvestris trees, acquiring 500 images that contained wood knots, dead knots, and checking defects. They achieved an average precision of 96.1% in detecting live knots, dead knots, and checking defects with the ensemble transfer learning SSD and DenseNet network.

Teo et al. [13] proposed automating timber quality control in the secondary wood industry by using an automated vision inspection system coupled with artificial intelligence. The authors have reviewed the machine learning and deep learning-based approaches regarding the identification of defects, discussed the contemporary algorithms and techniques, and further presented limitations and possible future research directions. Non-destructive testing technology urgently needs to be applied due to the excessive consumption of timber resources.
In the work by Mingyu et al. [14], they proposed a TL-ResNet34 deep learning model developed by combining ResNet-34 with transfer learning. This has contributed significantly to improving wood defect detection in the detection of wood defects caused by knots. With transfer learning, the detection accuracy was higher compared to other methods in the study and thus had a capability for better prediction in the detection of wood defects.

In a study conducted in central Croatia, by Martina et al. [15] regarding the potential of multispectral WorldView-3 (WV-3) satellite images for classifying three major tree species in a mixed deciduous forest, a relatively high overall accuracy of 85% through pixel-based supervised classification and RF and SVM algorithms was achieved using WV-3 spectral characteristics alone. The introduction of GLCM texture features improves the accuracy further, with variance being one of the influential features from GLCM. The integration of spectral and textural features increased overall accuracy by 10% and 7% of RF and SVM classification approaches, respectively, showing the importance of incorporating texture features in the classification of tree species from satellite imagery.

In another class of research, Janne et al. [16] conducted a research study on the classification of tree species for large-scale forest monitoring, focusing on those species which are of economic importance in addition to their ecologically beneficial counterparts, based on hyper-spectral and LiDAR data. Moreover, its performance was tested and compared with other recent popular methods, and even methodologies that have been popularized for hyper spectral data using 3D Convolutional Neural Network. The best 3D-CNN performed in an 83 km² study area in the southern boreal zone of Finland with an F1-score of 0.91 for aspen and an overall accuracy of 87%, outperforming support vector machines and artificial neural networks. This work also applied some new interpretability procedures, such as occlusion and saliency maps, and delivered a tree species map for the presented best 3D-CNN, enabling applications within the fields of sustainable forestry and biodiversity conservation.

Fanyou et al. [17] also performed the study on some deep convolutional neural network architectures for identification of hardwood lumber species. It achieved an accuracy of 98.2% in classifying 11 common hardwood species, showing how powerful deep learning can be in automating wood processing systems.

Bhusnurmath & Doddamani [18] proposed a deep learning-based approach in classifying 50 tree species using Bark texture images based on BarkVN-50 dataset - largest of its kind. Bark is an important attribute for identification of species as it remains persistent and structurally unique across seasons. The investigation compared feature extraction from basic model CNN with pre- trained models VGG16 and MobileNet. Results have shown that pre-trained models outperform the simple CNNs with a very large margin regarding accuracy and computational efficiency in most forestry-related tasks, including conservation, disease diagnosis, and plant production.

Bhusnurmath & Doddamani [19][23] Proposed machine learning algorithms to classify the texture images by feature extraction techniques for that they used the haralick feature and from that features created the CSV file and applied the different machine learning algorithms. Calculated result of all models. And later for lower resulted models are ensembles together with the voting classifier and calculate the results.

3. PROPOSED METHOD

Following the literature review, it was observed that incorporating advanced mechanisms like Bottleneck Attention can significantly enhance model performance and feature representation, which informs the proposed method in this section.

Figure 1 represents the adopted research methodology, and it gives an abstract of the technique followed for the classification of various wood defects and normal wood surface images. The steps involved in the proposed methodology are as follows:

Step 1: Downloaded a large-scale image dataset of wood surface defects [20][21], which includes 4,000 images across 8 classes.

Step 2: Collected normal wood surface images from another texture dataset [22], which consists of the different kinds of texture images of 15 classes.

Step 3: For the experiment combined these two datasets to form a new dataset that is new **merged dataset** comprising 7 different classes.

Step 4: For the merged dataset, various generalized pre-processing techniques were applied.

Step 5: Further used the different deep learning models to classify the 7 different classes.

Step 6: Finally evaluated the performance of each models best results.

Below, Figure 1 illustrates the workflow for the proposed experiment, detailing each step from dataset preparation to model training and evaluation.

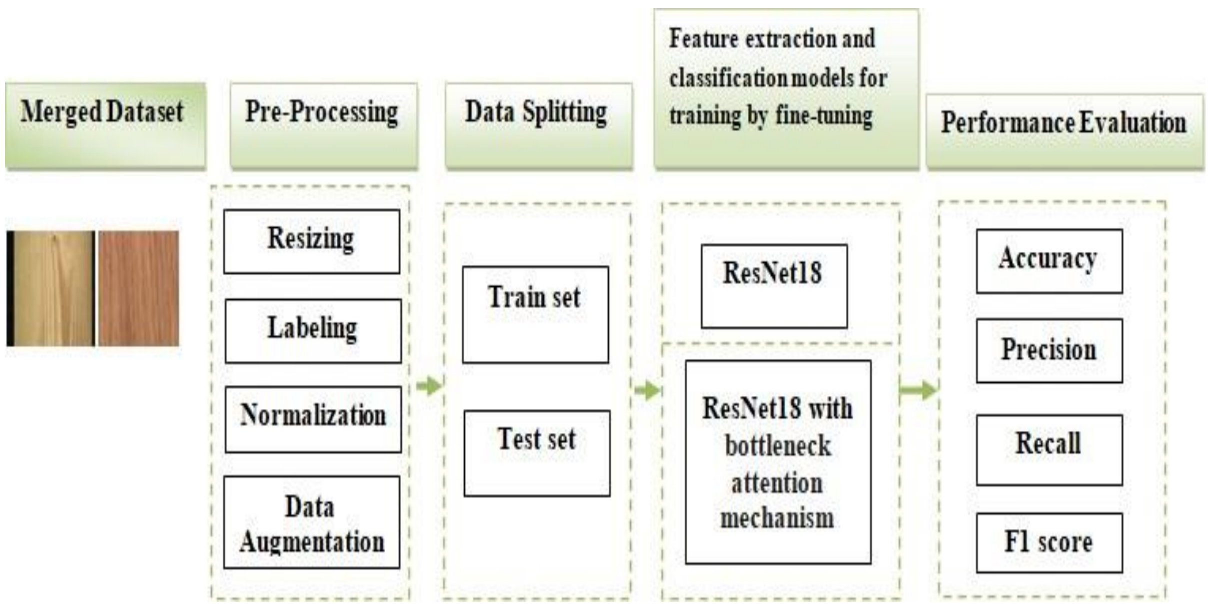


Figure 1: Work flow diagram of the proposed study

3.1. Data collection

The first benchmark dataset used in this study was downloaded from Kaggle named "**Large Scale Image Dataset of Wood Surface Defects"** [20][21]. The original dataset comprises high-resolution images captured with specialized cameras, with each image requiring approximately 12 megabytes (MB) of storage space. Facilitating accurate defect localization and classification.

The second dataset used in this study was also downloaded from the kaggle "Wood Surface" [22]. The original dataset consists of the different kinds of surfaces like Steel, Carpet, Capsules, Grids, Tiles, wood etc….From these many of classes, for the experiment collected the only wood surface images.

3.2. Data preparation

Dataset 1: From the dataset 1 among 4,000 images of 8 classes and accompanied by an annotation file from Kaggle, for the experiment selected images belonging to 6 specific classes for proposed work. Authors used a total of 272 images due to the following reasons: many images lacked proper annotations, and numerous images contained combinations of two or more defects. The study focused on six classes, excluding two classes with very low image counts.

Figure 2 and 3 shows the original dataset sample images with one sample annotation file.

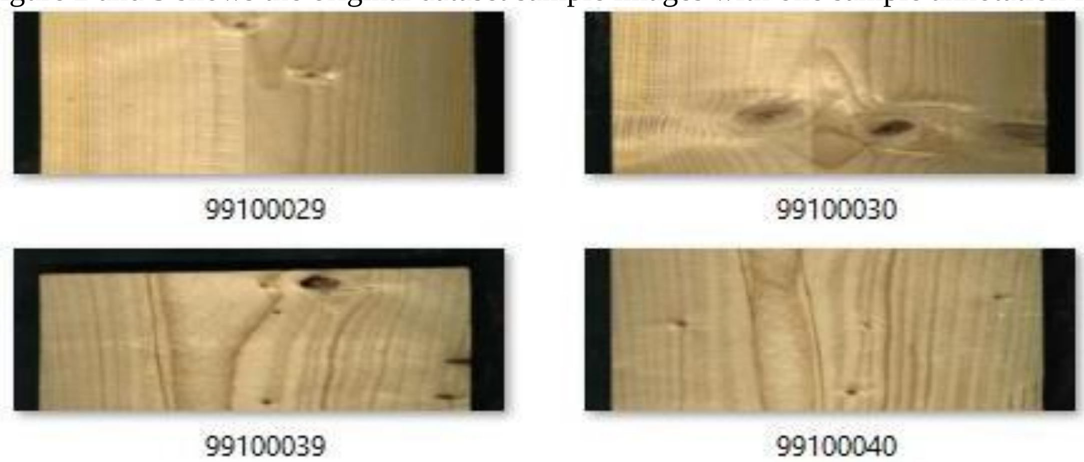


Figure 2: Sample original images from the dataset 1.

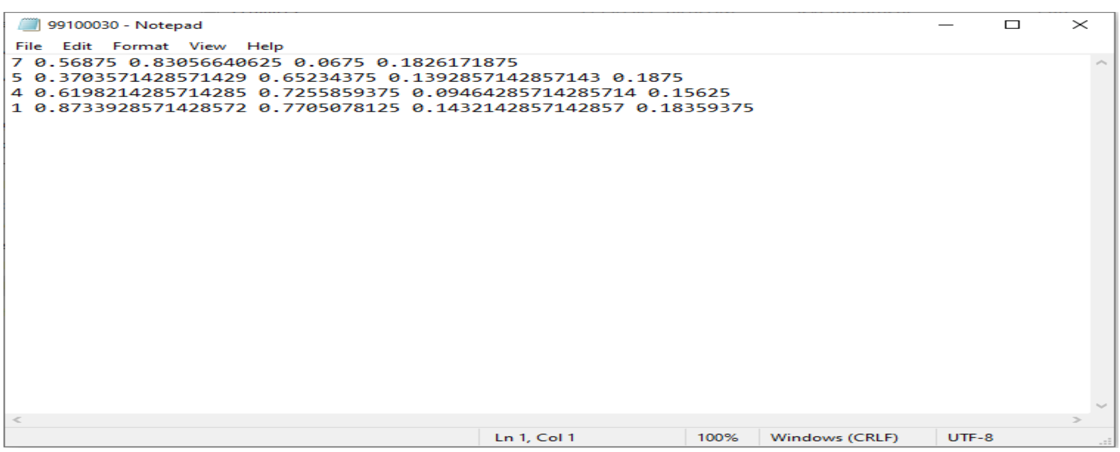


Figure 3: Sample of annotation file provided with the original dataset 1.

From the Figure 3 it is observed that the annotation file for the one image named “99100030” which belongs to 7, 5, 4, and 1. I.e. the classes are Crack, knot_with_crack, Dead_Knot and Live_Knot respectively. These kinds of images which consists the more than one defects are left for the proposed experiment.

Using the annotation file, 272 images were categorized into 6 distinct classes. The resulting, modified Dataset 1 now contains these 272 images organized into the 6 classes. The selected classes are: CRACK, DEAD_KNOT, KNOT_WITH_CRACK, LIVE_KNOT, MARROW, and RESIN.

Table 1 shows the details about the original dataset 1 and modified dataset 1 with total number of images and class wise distribution of images also shows the images size.

Table 1. Description of dataset 1 considered for the proposed experimentation.

ORIGINAL DATASET 1		MODIFIED DATASET 1	
Number of Images	**Images size**	**Number of Images**	**Image size**
4000	2800 X 1024	272	224 X 224
4000 Images with 4000 annotation file (Text file)		**FORMED CLASSES**	
		Classes	**Number of Images**
		Crack	42
		Dead_knot	48
		Knot_with_crack	38
		Live_knot	50
		Marrow	51
		Resin	43

Dataset 2: And from the dataset 2 among several classes, for the experiment chosen the one class called ‘Wood surface images’ normal images which consists of the 281 images.

Table 2. Description of dataset 2 considered for the proposed experimentation.

ORIGINAL DATASET 2		SELECTED CLASS (DATASET 2)		
Number of classes	**Images size**	**Number of classes**	**Converted Images size**	**Number of Images**
15	1024 X 1024	01-Wood	224 X 224	281

Table 2 shows the details about the dataset 2 which is considered for the experiment. The original dataset consists of the different kinds of surfaces texture images of 15 classes like Steel, Carpet, Capsules, Grids, Tiles, wood etc….From 15 classes, for the experiment collected the only wood surface images which consists the 281 images and these images are normal images considered for the proposed experiment.

Finally dataset 1 and dataset 2 are merged to form new merged dataset for the proposed experiment. Overall the experimented **merged dataset** consists of the 553 images with 7 classes including the normal and defected classes in it. A detail about the prepared **merged dataset** for the experiment is shown in Table 2.

The Figure 4 shows the flow diagram of merging two different dataset to form the standard **merged dataset** which consists of the both normal and defects classes in it.

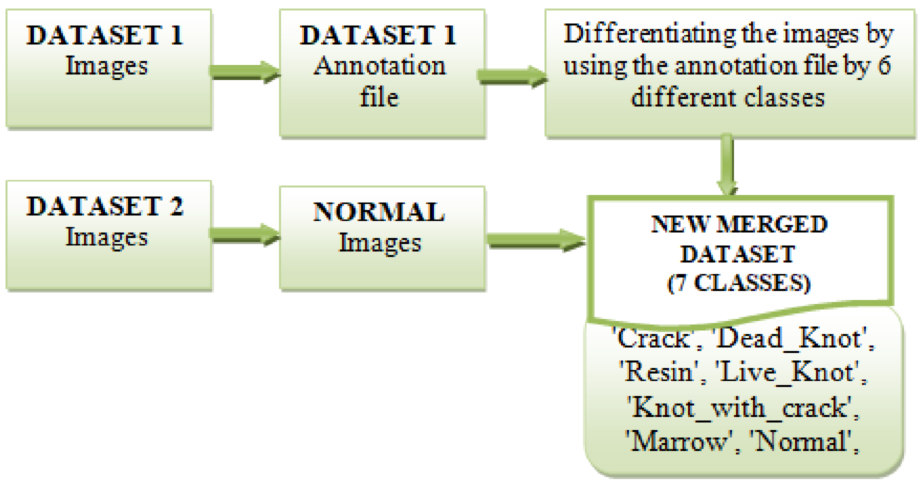


Figure 4: Flow diagram for the merged dataset preparation

The Table 3 describe the in depth overview of the merged dataset after preparation**.**

Table 3. Description of merged dataset considered for the proposed experimentation.

Merged dataset (Dataset 1 + Dataset 2)	
Parameters	***Values***
Number of images	553
Number of Images after data augmentation	(No. of epochs) x (No. of Images) Dynamically
Number of classes	7
Image size	224 x 224
Image format	RGB JPG

From the Figure 2 and Table 3 it is observed that the final merged dataset preparation process and details about the merged dataset .And both the datasets images were of different sizes initially, In pre-processing step all image size is standardized size 224 X 224.

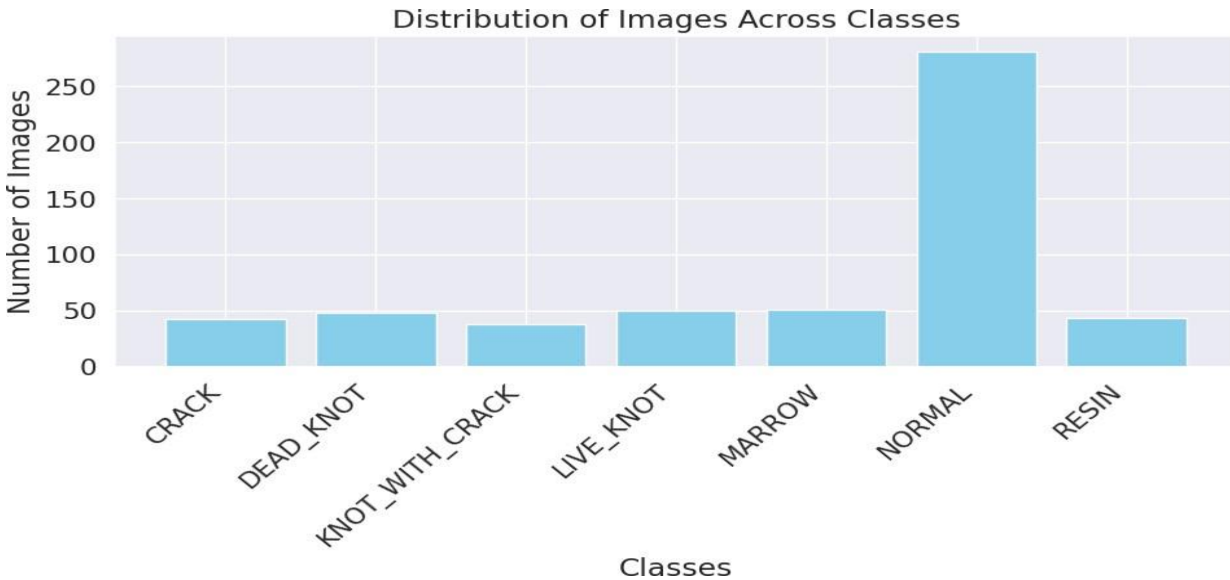


Figure 5: Detailed view of classes in merged dataset.

The Figure 5 represents each class along with the varying number of images contained within each class, highlighting the extent of this class imbalance.

3.3. Data pre-processing

For the merged dataset authors have applied data preprocessing techniques to ensure the quality and consistency of the images before feeding them into the deep learning models. The preprocessing steps are listed below.

Resizing: All images were resized to 224x224 pixels to ensure uniformity, suitable for ResNet18 model input, facilitating efficient batch processing.
Data Augmentation: Techniques such as random cropping, horizontal and vertical flipping, rotation, and adjustments to brightness and contrast are used to enhance model robustness and prevent over fitting.
Class Label Inclusion: Renamed images to include class labels in filenames for easy identification and management.
Normalization: Pixel values were scaled to a range of 0 to 1 or normalized based on the ImageNet dataset mean and standard deviation. This step aids in speeding up convergence during model training.
Figure 6 shows the images of merged dataset after pre-processing and from the Figure 4 it is observed that the labeling of image names are arranged in serial wise before it was according to the annotation file and size of all images were resized to unique for all images of all classes.

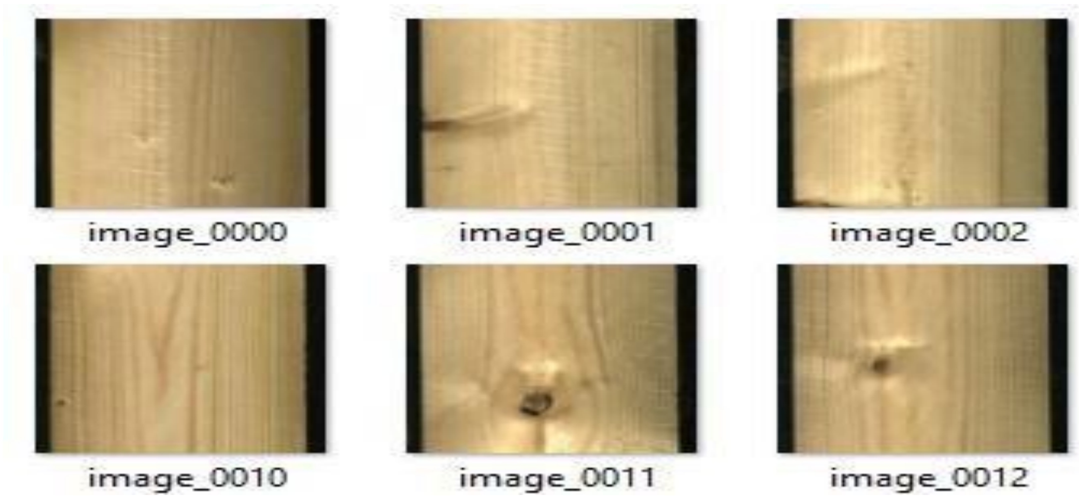


Figure 6: Images from the merged dataset after pre-processing.

3.4 Dataset splitting

The dataset is partitioned into two subsets: the training set, which encompasses 80% of the data, and the test set, which comprises the remaining 20%. This division ensures an effective balance between model training and evaluation.

3.5 Model building

After completing the dataset preparation up to section 3.4, the authors worked with ResNet18 and ResNet18 with BAM to evaluate the impact of the Bottleneck Attention Mechanism on model performance and feature representation. By exploring the dataset using both models, they gained unique insights that enriched the understanding of the data and guided the analysis towards more comprehensive conclusions.

3.5.1 ResNet18

Residual Blocks: The most fundamental novelty in ResNet is the proposal of residual blocks. These blocks utilize skip connections or shortcuts to jump over one or more layers. The residual block can be represented as in Equation 1.

$$y=F(x, \{ Wi \})+x \longrightarrow \qquad (1)$$

Where y is the output, x is the input, and F(x,{Wi}) is the residual mapping to be learned. This will help to avoid the vanishing gradient problem, since gradients would flow more easily through the network.

Skip Connections: Skip connections allow the input to skip some layers between its input and the output, and simply add them to the output of those layers. This makes it easy to train very deep networks by preserving the information and gradients.

In both cases, the formula is the same, but the residual block refers to the architectural unit incorporating the skip connection, while skip connections describe the mechanism itself. This is shown in the Equation 2.

$$y=F(x, \{Wi\}) +x \longrightarrow \qquad (2)$$

Depth: ResNet-18 consists of 18 layers, including convolutional layers, batch normalization layers, ReLU activations, pooling layers, and fully connected layers.

The Table 4 shows the optimized parameter and values used in the ResNet-18 model for the experiment.

Table 4: optimized parameter-values list of ResNet-18 model 1

Parameter	Optimized Values
Model Name	ResNet-18
Pre-trained	Yes (ImageNet)
Number of Layers	18
Loss Function	CrossEntropyLoss
Optimizer	SGD
Learning Rate (lr)	0.001
Momentum	0.9
Activation Function	ReLU
Pooling Layers	Max Pooling, Average Pooling
Convolutional Layer	Kernel Size: 3x3 (typical)
Batch Normalization	After each convolutional layer
Input Image Size	224 x 224

3.5.2 ResNet18 with Bottleneck Attention Mechanism

ResNet-18 architecture includes a Bottleneck Attention Mechanism (BAM) to improve its performance by focusing on important features in the input data. The BAM is integrated into the residual blocks of ResNet-18, enhancing the model's ability to concentrate on crucial information while suppressing less relevant features.

Bottleneck Attention Mechanism (BAM)

The Bottleneck Attention Mechanism is a channel-wise attention mechanism that enhances feature representation by focusing on important channels. It consists of:

- **Adaptive Average Pooling:** Reduces spatial dimensions to a vector of size equal to the number of channels as shown in equation 3.

$$Z_c = \frac{1}{H\ X\ W} \sum_{i=1}^{H} \sum_{j=1}^{W} Xc(i,j) \longrightarrow \quad (3)$$

Where Xc is the feature map of the c-th channel, H and W are the height and width of the feature map, and Zc is the pooled output for the c-th channel.

- **Fully Connected Layers:** Two layers reduce the dimensionality and then restore it, applying ReLU and Sigmoid activations to scale the importance of each channel.

Below The Table 5 shows the optimized parameter and values used in the ResNet-18 with BAM model

Table 5: Optimized parameter-values list of model 2 ResNet-18 with BAM

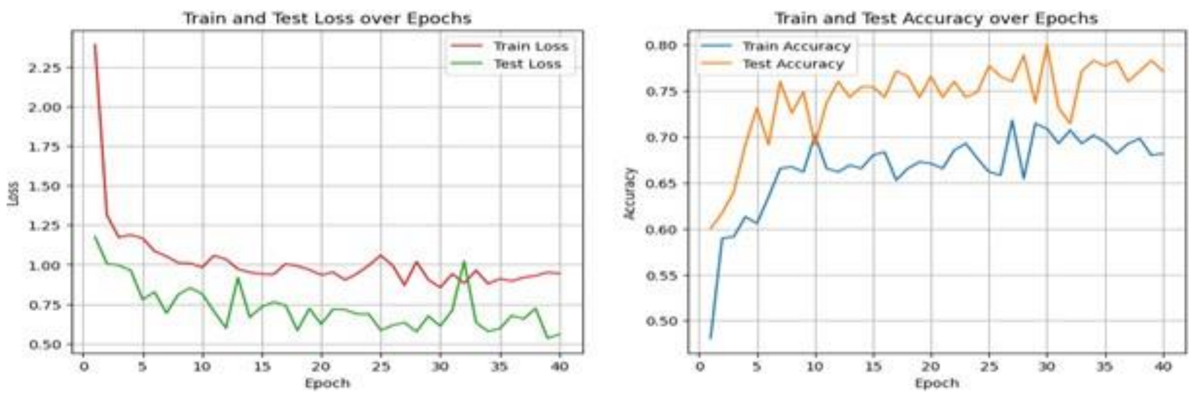
Parameter	Optimized Value
Model Name	Modified ResNet-18
Pre-trained	Yes (ImageNet)
Number of Layers	18
Bottleneck Attention	After each 3x3 convolutional layer
Convolutional Layers	Kernel Size: 3x3 (typical)
	Bottleneck Attention: Yes
Bottleneck Attention	Reduction Ratio: 16
Final Fully Connected Layer	in_features: 512
Loss Function	CrossEntropyLoss
Optimizer	SGD
Learning Rate (lr)	0.001
Momentum	0.9
Activation Function	ReLU
Pooling Layers	Max Pooling, Average Pooling
Batch Normalization	After each convolutional layer
Input Image Size	224 x 224

4. EXPERIMENTAL RESULTS

The machine used to run the research conducted by the authors was an Intel Core i3, 2.40 GHz processor with 4 GB of RAM, running a 64-bit Windows 10 Operating System. Python was used, particularly Numpy and Pandas libraries, while frameworks such as TensorFlow and PyTorch were used due to their GPU computing capability in training neural networks.

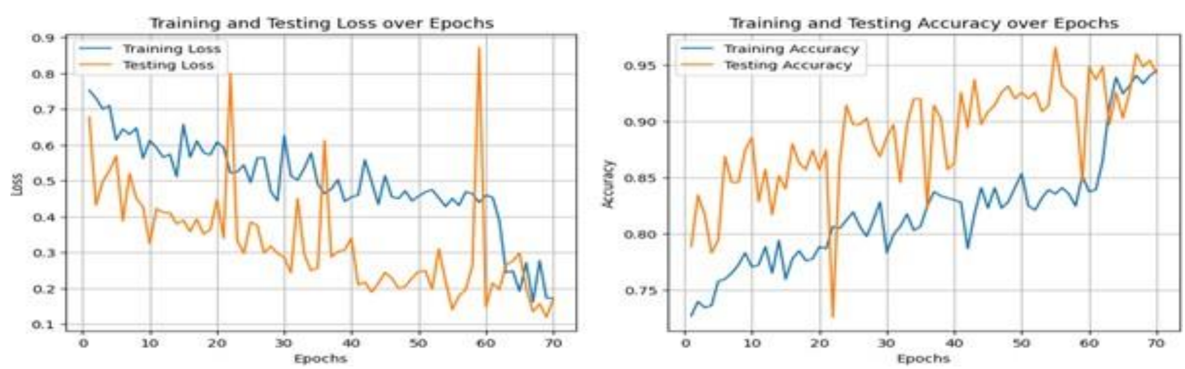
As discussed in the Section 3.5 the deep learning models were utilized for the proposed experiment with proper dataset and pre-processing of the datasets. This section offers a concise overview of the numerical results obtained, highlights discernible trends, and includes graphical illustrations derived from the experimental data.

The Figures 7 and 8 displays the detailed train, test loss curves and train, test accuracy curves for the ResNet18 model and ResNet18 with BAM respectively. Model is trained over 40-70 epochs. These curves offer a comprehensive view of the models' performance across the training process, capturing subtle variations in accuracy and loss.



(a) (b)

Figure 7: (a) Loss curves (b) Accuracy curves of ResNet18 model



(a) (b)

Figure 8: (a) Loss curves (b) Accuracy curves of ResNet18 with BAM model.

Figures 4 and 5 reveal that ResNet-18 with the Bottleneck Attention Mechanism (BAM) generally exhibits higher accuracy and lower loss compared to ResNet-18 without BAM, indicating that BAM enhances model performance by focusing on important features.

On top of that, ResNet-18 with the added BAM has performances improved over the ResNet-18 baseline in both metrics, further showing that deeper networks can effectively capture more complex patterns. The effectiveness of BAM constitutes better learning and generalization, thus thoroughly testifying to its value in performance optimization with different depths.

These variations, observed as good in variation, could be noted since the databases adopt high-class imbalance, which affects the model's ability to classify the underrepresented classes with such relevance.

Fig. 9 and 10 present the confusion matrix for the models.

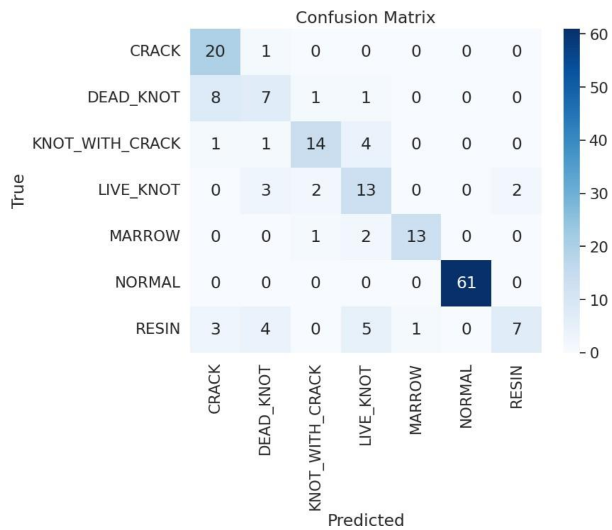


Figure 9: Confusion matrix of ResNet18 model.

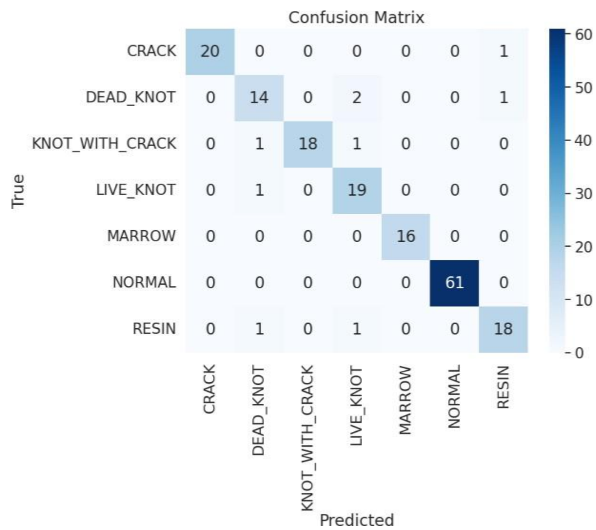


Figure 10: Confusion matrix of ResNet18 with BAM model.

Figures 9 and 10 reveal some interesting trends hidden in the confusion matrices with respect to the model performance on all 7 classes. The matrices place in perspective both correctly classified and misclassified instances of a class, where the NORMAL class has higher accuracy, while RESIN is mostly misclassified as DEAD_KNOT or LIVE_KNOT by both models. The model clearly shows in the matrix what kind of classes fail, in particular, those that look alike, such as different types of knots and cracks, which can be pretty insightful when it comes to improvement.

The classification report was also created for both models, giving a summary of precision, recall, and F1 score for all classes, and this further clarified the strengths of the models and projection areas of the model.

Table 6: classification report of model 1 ResNet18

Classes Parameters	**Precision**	**Recall**	**F1-Score**
Crack	0.62	0.95	0.75
Dead_knot	0.42	0.41	0.42
Knot_with_crack	0.78	0.70	0.74
Live_knot	0.52	0.65	0.58
Marrow	0.93	0.81	0.87
Resin	0.78	0.35	0.48
Normal	1.00	1.00	1.00

Table 6 also shows that Normal and Marrow classes are the best in terms of performance of this model, where both achieved perfect precision, recall, and F1-score. Even Crack and Knot_with_crack classes show a reasonable performance with pretty decent precision and F1-scores. However, this classifies Dead_knot and Resin as serious problems; the model is not efficient enough to capture these defectives. While Live_knot showed a mediocre performance, reaching balanced but unremarkable scores.

Table 7: classification report of model 2 ResNet18 with BAM

Classes / Parameters	Precision	Recall	F1-Score
Crack	1.00	0.95	0.98
Dead_knot	0.82	0.82	0.82
Knot_with_crack	1.00	0.90	0.95
Live_knot	0.83	0.95	0.88
Marrow	1.00	1.00	1.00
Resin	0.90	0.90	0.90
Normal	1.00	1.00	1.00

Table 7 shows that ResNet18 with BAM model did a really excellent job for most classes. For the classes Crack, Knot_with_crack, Marrow, and Normal, the model got perfect precision and recall, making the F1 score close to or at 1.00. Dead_knot and Resin are also doing great, where both the precision and recall were well balanced at 0.82 and 0.90, respectively. It feeds live_knot with high recall and good precision, resulting in a solid F1-score of 0.88. Overall, the model ResNet18 with BAM is good in all classes.

Overall performance is good in ResNet18, but it gives poor performances on classes Dead_knot and Resin. On the other side, ResNet18 with BAM significantly outperformed others with close-to-perfect scores on all classes.

The authors experimented with different optimizers: from Adam and RMSprop to SGD later on, in order to comprehensively perform an evaluation of the effect of each one of them on model training dynamics and general model performance. The authors explored multiple optimization strategies in pursuit of the best combination that might offer maximum accuracy with efficiency.

Table 8: The following table presents a detailed model performance comparison, done by different metrics arising from training various optimizers.

Table 8: Comparative analysis of model performance with various optimizers

Models	Evaluation matrices			
	Accuracy	**Precision**	**Recall**	**F1-score**
	SGD Optimizer			
ResNet18	77.14	78.82	77.14	76.50
ResNet18 with BAM	94.86	95.16	94,86	94.92
	Adam Optimizer			
ResNet18	72.90	73.24	72.43	70.14
ResNet18 with BAM	81.31	81.74	81.31	79.72
	RMSprop Optimizer			
ResNet18	65.89	77.57	65.89	68.44
ResNet18 with BAM	80.37	85.64	80.37	78.27

From Table 8, it can be seen that the results for ResNet18 vary from one optimizer to another but its metrics improve once BAM is used. With an optimizer as simple as SGD, ResNet18 yields an accuracy of 77.14%. On the other hand, for this very network setting, application of the BAM has increased it significantly to 94.86%. For an Adam optimizer, the BAM increases the accuracy from 72.90% to 81.31% in case of an Adam optimizer. In the RMSprop optimizer, it gives an accuracy change from 65.89% to 80.37% when BAM is added. These results hint the fact that the BAM module boosts the performance of the ResNet18—seriously—for all optimizers; the best one is done by the SGD optimizer.

Table 8 depicts the accuracy, precision, recall, and F1 score comparisons of both models, which were trained using different optimizers. All these collectively depict how much more effective the selection of optimizers made the working model for the classification task while choosing the optimal configuration for diverse applications.

5. CONCLUSION

The research work reported here dealt with some serious wood industry challenges concerning the surface defect detection and classification, which are even more difficult due to variability in the raw material and manufacturing processes. Traditional inspection approaches are manual in nature, hence highly labor-intensive and subjective, and prone to inconsistencies. The authors have therefore built a new dataset including normal as well as defective wood surface images, in developing a way of surmounting these issues. It represents a foundational dataset to train and test the deep learning model adapted for the classification of defects on wood surfaces.

Experimental results showed the ResNet18 architecture with a bottleneck attention mechanism that was able to achieve an accuracy of 94.86% from a starting point of 77.14%. The author also used different optimizers like SGD, RMSprop, and Adam optimization algorithms, which proved very critical in the model performance for defect detection tasks.

REFERENCES

[1] Abdullah, N. D., Hashim, U. R. A., Ahmad, S., & Salahuddin, L. (2020). Analysis of texture features for wood defect classification. *Bulletin of Electrical Engineering and Informatics*, *9*(1), 121-128. DOI: https://doi.org/10.11591/eei.v9i1.1553

[2] Kodytek, P., Bodzas, A., & Bilik, P. (2021). A large-scale image dataset of wood surface defects for automated vision-based quality control processes. *F1000Research*, *10*. doi: 10.12688/f1000research.52903.2

[3] Bhusnurmath, R. A., & Hiremath, P. S. (2020). Anisotropic Diffusion-Based Color Texture Analysis for Industrial Application. In *Applications of Advanced Machine Intelligence in Computer Vision and Object Recognition: Emerging Research and Opportunities* (pp. 30- 64). IGI Global. DOI: 10.4018/978-1-7998-2736-8.ch002

[4] Aggarwal, A., & Kumar, M. (2021). Image surface texture analysis and classification using deep learning. *Multimedia Tools and Applications*, *80*(1), 1289- 1309.https://doi.org/10.1007/s11042-020-09520-2

[5] Xi, Z., Hopkinson, C., Rood, S. B., & Peddle, D. R. (2020). See the forest and the trees: Effective machine and deep learning algorithms for wood filtering and tree species classification from terrestrial laser scanning. *ISPRS Journal of Photogrammetry and Remote Sensing*, *168*, 1-16. https://doi.org/10.1016/j.isprsjprs.2020.08.001

[6] Liu, S., Jiang, W., Wu, L., Wen, H., Liu, M., & Wang, Y. (2020). Real-time classification of rubber wood boards using an SSR-based CNN. *IEEE Transactions on Instrumentation and Measurement*, *69*(11), 8725-8734. 10.1109/TIM.2020.3001370

[7] Souza, D. V., Santos, J. X., Vieira, H. C., Naide, T. L., Nisgoski, S., & Oliveira, L. E. S. (2020). An automatic recognition system of Brazilian flora species based on textural features of macroscopic images of wood. *Wood Science and Technology*, *54*(4), 1065-1090. https://doi.org/10.1007/s00226-020-01196-z

[8] 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*, 105941. https://doi.org/10.1016/j.compag.2020.105941

[9] Lim, J., Kim, K. M., Kim, E. H., & Jin, R. (2020). Machine learning for tree species classification using sentinel-2 spectral information, crown texture, and environmental variables. *Remote Sensing*, *12*(12), 2049. **https://doi.org/10.3390/rs12122049**

[10] Haq, M. A., Rahaman, G., Baral, P., & Ghosh, A. (2021). Deep learning based supervised image classification using UAV images for forest areas classification. *Journal of the Indian Society of Remote Sensing*, *49*(3), 601-606. https://doi.org/10.1007/s12524-020-01231-3

[11] Diez, Y., Kentsch, S., Fukuda, M., Caceres, M. L. L., Moritake, K., & Cabezas, M. (2021). Deep learning in forestry using uav-acquired rgb data: A practical review. *Remote Sensing*, *13*(14), 2837. **https://doi.org/10.3390/rs13142837**

[12] Ding, F., Zhuang, Z., Liu, Y., Jiang, D., Yan, X., & Wang, Z. (2020). Detecting defects on solid wood panels based on an improved SSD algorithm. *Sensors*, *20*(18), 5315. **https://doi.org/10.3390/s20185315**

[13] Chun, T. H., Hashim, U. R. A., Ahmad, S., Salahuddin, L., Choon, N. H., & Kanchymalay, K. (2023). A review of the automated timber defect identification approach. *International Journal of Electrical & Computer Engineering (2088-8708)*, *13*(2). DOI: **10.11591/ijece.v13i2.pp2156-2166**

[14] Gao, M., Qi, D., Mu, H., & Chen, J. (2021). A transfer residual neural network based on ResNet-34 for detection of wood knot defects. *Forests*, *12*(2), 212. **https://doi.org/10.3390/f12020212**

[15] Deur, M., Gašparović, M., & Balenović, I. (2020). Tree species classification in mixed deciduous forests using very high spatial resolution satellite imagery and machine learning methods. *Remote Sensing*, *12*(23), 3926. **https://doi.org/10.3390/rs12233926**

[16] Mäyrä, J., Keski-Saari, S., Kivinen, S., Tanhuanpää, T., Hurskainen, P., Kullberg, P., ... & Vihervaara, P. (2021). Tree species classification from airborne hyperspectral and LiDAR data using 3D convolutional neural networks. *Remote Sensing of Environment*, *256*, 112322. https://doi.org/10.1016/j.rse.2021.112322

[17] Wu, F., Gazo, R., Haviarova, E., & Benes, B. (2021). Wood identification based on longitudinal section images by using deep learning. *Wood Science and Technology*, *55*, 553-563. https://doi.org/10.1007/s00226-021-01261-1

[18] Bhusnurmath, R. A., & Doddamani, S. (2023, June). Bark Texture Classification Using Deep Transfer Learning. In *International Conference on Multi-disciplinary Trends in Artificial Intelligence* (pp. 407-420). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-36402-0_38

[19] Bhusnurmath, R. A., & Doddamani, S. (2023, February). Texture Feature Extraction and Classification Using Machine Learning Techniques. In *International Conference on Emerging Research in Computing, Information, Communication and Applications* (pp. 509- 520). Singapore: Springer Nature Singapore. https://doi.org/10.1007/978-981-99-7622-5_35

[20] Bhusnurmath, R. A., & Doddamani, S.(2023) Wood Surface Defect Detection Using Discrete Wavelet Transform and Deep-Net Model, In **3rd International Conference on Advancements in Smart Computing and Information Security (ASCIS - 2024)** https://doi.org/10.35444/IJANA.2024.15429

[21] https://www.kaggle.com/datasets/nomihsa965/large-scale-image-dataset-of-wood-surface- defects

[22] https://www.kaggle.com/code/miladaghalari/wood-defect-localization/input

[23] Bhusnurmath, R. A., & Doddamani, S. ENSEMBLE MACHINE LEARNING MODEL FOR TEXTURE FEATURE EXTRACTION AND CLASSIFICATION, in International Journal of Applied Engineering & Technology Vol. 5 No.4, December, 2023 https://romanpub.com/resources/ijaet20v5-4-2023-339.pdf