WoodYOLO: A Novel Object Detector for Wood Species Detection in Microscopic Images

Anonymous ECCV 2024 Submission

Paper ID #3

Abstract. Wood species identification plays a crucial role in various industries, from ensuring the legality of timber products to advancing ecological conservation efforts. This paper introduces WoodYOLO, a novel object detection algorithm specifically designed for microscopic wood fiber analysis. Our approach adapts the YOLO architecture to address the challenges posed by large, high-resolution microscopy images and the need for high recall in localization of the cell type of interest (vessel elements).

Our results show that WoodYOLO significantly outperforms state-of-the-art models, achieving performance gains of 12.9% and 6.5% in F2 score over YOLOv10 and YOLOv7, respectively. This improvement in automated wood cell type localization capabilities contributes to enhancing regulatory compliance, supporting sustainable forestry practices, and promoting biodiversity conservation efforts globally.

Keywords: Object Detection · Microscopic Imaging · Forest Protection

1 Introduction

Global deforestation is a cause of biodiversity loss and climate change. The European Union's recently adopted EU Deforestation Regulation (EUDR, [25]), which replaces the EU Timber Regulation (EUTR), requires that products traded in the EU are based on deforestation-free supply chains. This increases the demand for confirming the declaration of wood products regarding the wood species and origin.

This is a particular challenge for paper products where the DNA is destroyed and different pulps are mixed during production. Therefore, neither genetics, stable isotopes nor NIR spectroscopy can be used. It is therefore not possible to analyze the origin [32, 37]. Recently a new, very complex chemotaxonomic method for determining wood species was introduced for the first time [7]). But the standard analysis for checking the declared wood species in paper is still the anatomy [12, 15]. After sample preparation, the microscopic examination of the cell characteristics by experts is time-consuming and requires a high level of personal experience. The limited number of experts in the field makes it challenging to meet the increasing demand for wood species verification [31].

To address these challenges, recent advancements in computer vision and machine learning offer promising avenues for automating wood species identification. Machine learning techniques, particularly deep neural networks, have shown

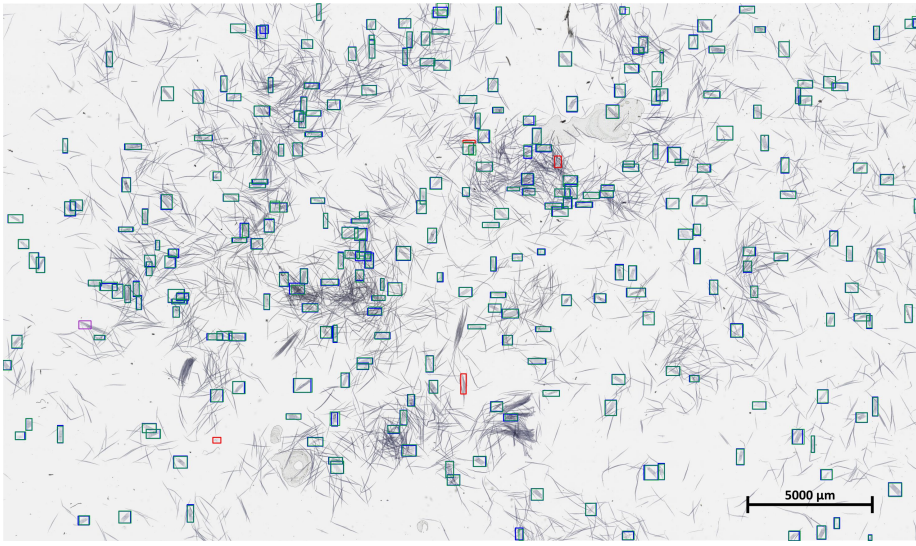

Fig. 1: Microscope image of macerated hardwood cells including vessel elements. Green boxes indicate ground truth annotations by wood anatomists, blue boxes represent predicted vessel locations, the purple box indicate one false negative and red boxes denote false positives that were not annotated by wood anatomists. WoodYOLO significantly speeds up the manual annotation process by automatically identifying hundreds of vessel elements.

remarkable capabilities in analyzing large-scale image datasets and extracting intricate features crucial for species classification [33]. However, while automated systems exist for macroscopic wood analysis [26, 38, 44], automated methods for microscopic analysis of fibrous materials like paper are still nascent [23].

A deep learning-based approach specifically targeting the detection and classification of vessel elements in microscopic images of macerated wood samples was recently presented [23]. These efforts have highlighted the potential of automation to streamline what has traditionally been a manual task. However, existing methods often face challenges such as suboptimal recall and high demands on computational power, especially when processing large and high-resolution microscopic images.

To address these limitations and further advance automated wood species identification, we present WoodYOLO, a novel object detection algorithm specifically designed for microscopic wood fiber analysis. WoodYOLO builds upon the YOLO (You Only Look Once) architecture, incorporating tailored optimizations to enhance performance in high-resolution microscopy.

Our algorithm introduces several key innovations:

- Customized YOLO-based architecture specifically optimized for microscopic images, achieving significant performance gains over YOLOv10 and YOLOv7 by 12.9% and 6.5% respectively in terms of F2 score, while using less VRAM.

- Introduction of a novel anchor box specification method, where users define only the maximum width and height of objects. This approach improves F2 score by 0.7%.
- Comprehensive evaluation of various architectural decisions in modern object detectors. Our findings reveal that optimizations designed for general datasets like COCO [18] may not always translate to improved performance in real-world datasets or different domains.

By advancing automated wood species identification capabilities, our work contributes to enhancing regulatory compliance, supporting sustainable forestry practices, and promoting biodiversity conservation efforts globally. WoodYOLO represents a significant step towards developing scalable, reliable and efficient methods for wood species identification in microscopic images of fibrous materials.

2 Related Work

The automated identification of wood species in microscopic images of fibrous materials has gained significant attention in recent years. This interest is driven by the need for efficient and accurate methods to support global wood fiber product controls.

A pioneering approach for the identification of hardwood species in microscopic images using deep learning techniques was introduced by [23]. They developed a methodology for generating a large dataset of macerated wood references, focusing on nine hardwood genera. This approach utilized a two-step process: first, detecting vessel elements using YOLOv7 [41], and then classifying these elements using convolutional neural networks (CNNs).

While the localization of objects achieved promising results, there remains room for improvement. Recently developed object detection algorithms, particularly those based on transformers, such as the DETR (DEtection TRansformer) model family [3, 24, 47, 48], have shown potential. However, they have not seen widespread use due to higher time complexity, slower training speeds or lower mAP on real-world datasets.

Another line of research is the continuation of YOLO. It is important to note that a higher version number in YOLO does not necessarily indicate an improvement; instead, different techniques are applied, which may or may not work on particular datasets. Since the original YOLO publication [27], only YOLOv2 [28] and YOLOv3 [29] were developed by the original authors. Other versions have been introduced by different institutes or companies, including YOLOv4 [2], Scaled-YOLOv4 [40], YOLOX [9], YOLOv6 [16], DAMO-YOLO [46], YOLOv9 [43], YOLOv10 [39], PP-YOLO [20], PP-YOLOv2 [14], and PP-YOLOE [45]. Notably, YOLOv5 and YOLOv8 have never been published. In our method section, we will analyze some of the different components found in these papers.

In most practical machine learning research and data competitions, YOLO remains the state-of-the-art (SOTA). Therefore, our focus is on developing an

object detector based on this literature. Our current work builds upon these foundations by introducing a novel object detection algorithm specifically tailored for vessel element detection in microscopic images of fibrous materials. By designing our detection algorithm with this task in mind, we can make better optimizations and avoid focusing on general-purpose detection datasets such as COCO.

Although there are numerous papers in the microscopy and satellite imaging literature that adapt YOLO for high-resolution image analysis, they generally rely on the original YOLO code base and make only minor changes. For example, [1, 21] adapted YOLOv5 for cell counting. There are also various studies in the field of satellite images in which YOLO [17,22] has been slightly modified. As a result, the improvements compared to the baseline are often only marginal. In contrast, we have developed our version of YOLO from scratch and tested components from different versions. This allows for more significant and customized improvements specifically for our application.

3 Method

Our detection framework is tailored to localize vessel elements in microscopic images, a crucial step for automating hardwood species identification in fibrous materials. We adapted the YOLO architecture for this domain, addressing the challenges posed by large image sizes (up to 54,000 x 31,000 pixels) and the need for high recall. Unlike algorithms such as DETR, which do not scale well for very large images and have slower training times, YOLO has proven effective in real-world applications, making it a suitable choice for our task.

Although the YOLO family includes various models optimized for general datasets like COCO, these models are not directly applicable to our problem due to their design for multiple classes and general-purpose images. Therefore, we customized YOLO by integrating components from different versions to optimize it for vessel detection without the need for classification.

In this section, we describe our model's architecture, loss function, metric, and additional approaches evaluated to enhance detection performance.

3.1 Architecture

Our model architecture begins with selecting a backbone capable of efficiently extracting features from large microscopic images. The backbone processes the input to generate multi-scale feature maps. We tested several backbones such as VGG11 [34], ConvNext [19], and ResNet [10], and combined their feature maps through a component known as the neck, which outputs three feature maps. Although more than three feature maps can be used, our evaluation showed no significant advantage in doing so.

Our neck architecture is based on YOLOv7-tiny. We also tested YOLOX's CSPNet [42] but found the former to be better. The use of a smaller architecture is due to the need for memory efficiency. Since we want to train the network

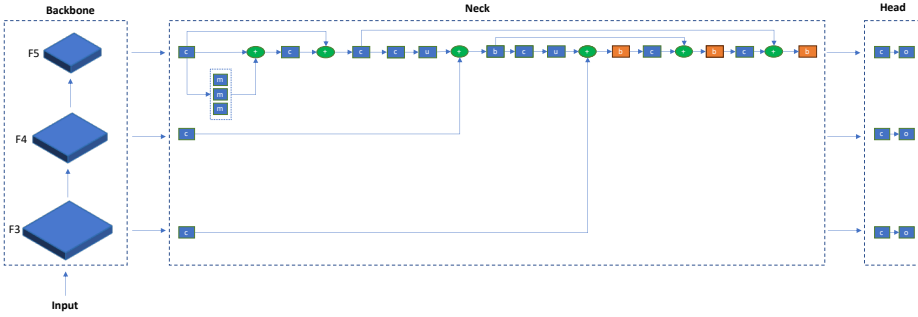

Fig. 2: Detection architecture based on YOLOv7-tiny [41]. "c" = Convolution with BN and ReLU, "+" = Concatenation, "m" = MaxPooling, "u" = Upsampling, "b" = Concatenation Block, "o" = single convolution with 5 outputs (x, y, width, height, confidence). Orange denotes an output in the neck of the model, which is given to the head. There are in total three outputs.

with a higher image resolution than the usual 640x640 or 1280x1280, we need to reduce the memory requirements. Also, deeper networks are usually chosen when many features are needed to distinguish between different classes. Here it is only a matter of finding objects without the need for classification. Therefore, simpler networks work better.

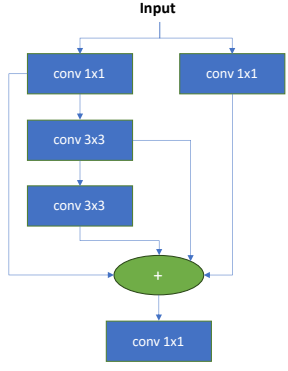

Fig. 3: The "b" concatenation block consists of convolutions of kernel size 3x3 and 1x1. Each convolution is followed by batch normalization and ReLU activation. The "+" means concatenation.

Fig. 2 shows that our neck consists of several convolutional layers that are combined in different ways. A "c" block consists of a simple convolution followed by a batch normalization and a ReLU function. The "b" block consists of parallel convolutions that are combined by concatenation. Fig. 3 shows the "b" block in detail.

The three orange blocks in Fig. 2 indicate the outputs of the neck. These three blocks are then used as inputs for the head. The outputs have different dimensions as a higher stride size is used for some of the convolutions.

The head produces the predictions of the neural network. It consists of only one convolutional block and one output convolution. A decoupled head, such as the one used in YOLOX, has not proven to be better in our case.

For each feature map, the head produces an output tensor f_i of dimensions $g_{h_i} \cdot g_{w_i} \times 5$, where g_{h_i} and g_{w_i} denote the grid height and width for the i-th layer. Each grid cell in f_i predicts five parameters: the center x-coordinate, center y-coordinate, width, height, and object confidence. These outputs are transformed as follows:

$$
\begin{aligned}
x_c &= 2\sigma(f_{i,\cdot,1}) - 0.5, \\
y_c &= 2\sigma(f_{i,\cdot,2}) - 0.5, \\
w &= \sigma(f_{i,\cdot,3})^2 \cdot g_{w_i} \cdot m_w, \\
h &= \sigma(f_{i,\cdot,4})^2 \cdot g_{h_i} \cdot m_h, \\
o &= \sigma(f_{i,\cdot,5}),
\end{aligned}
$$

where m_w and $m_h \in [0, 1]$ are hyperparameters defining the maximum width and height of the object. For instance, $m_w = 0.1$ means an object can be at most 10% of the total image width. This is similar to having a single anchor box of a maximum specific size.

The advantage of using two hyperparameters instead of anchor boxes is that no techniques such as clustering [28] have to be used to determine them. In addition, the loss function is much simpler and the training speed is higher.

The sigmoid function $\sigma(\cdot)$ ensures x_c and y_c are offsets within the grid, while w and h define bounding box dimensions. The confidence score o indicates the likelihood of a bounding box's presence at each location.

x_c and y_c are scaled here between $[-0.5, 1.5]$. This allows the model to shift the center of the box half to the left or right.

In the prediction phase, x_c and y_c offsets are adjusted by adding the grid indices $\{0, 1, \ldots, g_w\}$ and $\{0, 1, \ldots, g_h\}$, respectively. Coordinates are scaled to the original image size by multiplying x_c, y_c, w, and h by $\frac{s_h}{g_{h_i}}$ and $\frac{s_w}{g_{w_i}}$, where s_h and s_w are the input image dimensions.

3.2 Loss Function

Our loss function consists of two components:

$$
L = L_r + L_p \,,
$$

where L_r is the regression loss and L_p is the classification loss.

Regression Loss The regression loss measures the alignment between predicted bounding boxes $\hat{b}$ and ground truth b using the Intersection over Union (IoU):

$$L_r = \frac{1}{m}\sum_{i=1}^{n}\sum_{j=1}^{m}(1 - \text{IoU}(\hat{b}_{i,j}, b_{i,j})),$$

where n is the number of feature pyramid layers (in our case, $n = 3$) and m is the number of bounding boxes. The regression loss is either evaluated with the corresponding bounding box at that grid cell or additionally with neighboring grid cells (multi-positives).

There are different variants of IoU: Complete IoU (cIoU) [50], Distance IoU (DIoU) [49], Generalized IoU (GIoU) [30] and standard IoU. In the evaluation section, we evaluate the different approaches to see which maximizes our metric.

Classification Loss The classification loss evaluates the confidence score $\hat{o}$ using binary cross entropy (BCE), with the ground truth confidence o derived from IoU:

$$L_p = \frac{1}{m}\sum_{i=1}^{n}\sum_{j=1}^{m}\text{BCE}(\hat{o}_{i,j}, \text{IoU}(\hat{b}_{i,j}, b_{i,j})).$$

Unlike the regression loss, we evaluate BCE at all locations of the grid. However, we set $\text{IoU}(\hat{b}_{i,j}, b_{i,j}) = 0$ when there is no ground truth box at a specific grid cell.

3.3 Metric

The predominant metric in object detection is average precision (AP) [5] computed at different thresholds, which summarizes both precision and recall:

$$\text{AP} = \int_0^1 p(r)\, dr\,,$$

where r denotes recall and $p(r)$ denotes precision as a function of recall. A detection is considered correct if the IoU between the predicted and the true bounding box exceeds a predefined threshold.

In our specific application, however, the use of AP would not be a good choice. Recall takes precedence over precision as our goal is to find all objects.

Furthermore, we are less interested in an exact overlap with the ground truth. Minor shifts or size variations in the bounding box should not be penalized by the metric. Therefore, we want to consider only a single low IoU threshold. Often AP is computed at multiple thresholds.

Hence, we propose an alternative metric: the F2 score, which is computed with a fixed IoU threshold of 0.3. This choice emphasizes recall over precision. False positives can be handled in a postprocessing step by training a classifier to distinguish between correct and wrong detections. We see in Fig. 4 two examples where the overlap of 30% is sufficient.

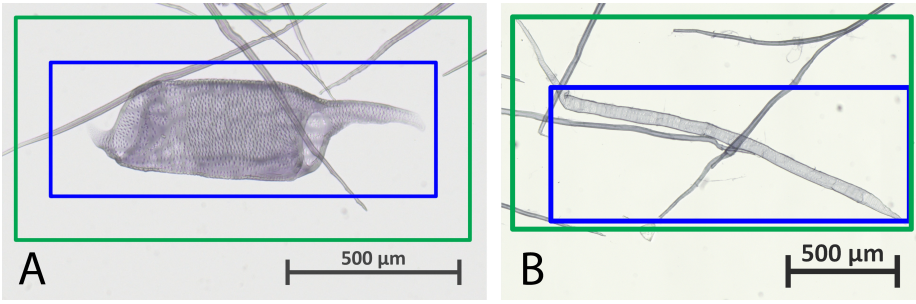

Fig. 4: Comparison of predicted bounding boxes (blue) and ground truth boxes (green). A high IoU threshold can result in both predicted boxes being rated as errors. **(A)** The overlap is below 0.5. Due to incorrect annotations, the predicted bounding boxes are sometimes more accurate. **(B)** Imperfect prediction as the end of the object (vessel element) is not detected.

While the usual threshold is 0.5, we choose a lower threshold of 0.3. This threshold takes into account the fact that perfect alignment with the ground truth bounding box is not essential for our objectives.

3.4 Additional Approaches

We explored several innovations from the YOLO series to further enhance our detection framework, evaluating their impact on performance. Some of these results will be shown in the evaluation section.

Center Sampling and Multi-Positives We explored the use of neighboring grid cells for matching ground truth boxes, a technique known in the literature as multi-positives [9] or center sampling [36].

In the standard loss function L_r, we compute the IoU loss only between boxes at coordinates (i, j). Center sampling extends this concept by also comparing boxes at $(i + k_1, j + k_2)$, where k_1 and k_2 are integer offsets. The ground truth box is duplicated for these new coordinates $(i+k_1, j+k_2)$ to make a comparison with the ground truth box at those positions possible. We investigated three variants:

$$\text{0 Neighbors: } \begin{bmatrix} 0 & 0 & 0 \\ 0 & \circ & 0 \\ 0 & 0 & 0 \end{bmatrix} \qquad \text{2 Neighbors: } \begin{bmatrix} 0 & \times & 0 \\ 0 & \circ & \times \\ 0 & 0 & 0 \end{bmatrix} \qquad \text{4 Neighbors: } \begin{bmatrix} 0 & \times & 0 \\ \times & \circ & \times \\ 0 & \times & 0 \end{bmatrix}$$

Here, $\circ$ denotes the original bounding box, while $\times$ represents neighboring boxes and 0 means "empty cell". For the 0 neighbors configuration, the loss L_r remains unchanged as it only considers the original box. In the 2 neighbors configuration, the nearest bounding boxes within the grid are selected, in this case, the right and upper boxes. For the 4 neighbors configuration, we use bounding

boxes from all directions: left, right, up, and down. Note that the diagonal boxes are never selected.

Since object detection is a one-to-many mapping (one ground-truth box corresponds to many correctly predicted boxes), this strategy attempts to simulate this mapping using the loss function.

Label Assignment Bounding boxes are predicted for every feature map. The use of center sampling further increases the number of predicted boxes. To manage this increase of bounding boxes, we evaluated label assignment strategies designed to reduce the number of valid boxes per object.

We experimented with modern label assignment techniques such as SimOTA and TAL [6, 8]. However, these methods did not yield improved results in our scenario. We attribute this to our metric, which prioritizes maximizing recall rather than balancing precision and recall.

Auxiliary Head Loss Deep supervision techniques, such as those used in YOLOv7 [41], involve adding auxiliary losses to guide deeper networks. Our experiments with additional model layers showed no benefit, so this approach was excluded from our final model.

Anchor Boxes Anchor boxes, introduced in YOLOv2 [28], are used to predict object locations. Consistent with YOLOX findings [9], our results showed no improvement with anchor boxes, leading us to exclude them for simplicity. Instead, we incorporate parameters m_h and m_w in the range $[0, 1]$ to constrain the predicted width and height of bounding boxes, as discussed previously.

NMS-Free Detection NMS-free approaches from models like YOLOv10 did not perform as well in our tests. We retained traditional Non-Maximum Suppression (NMS) for its robustness and simplicity.

Training Strategies Techniques such as mosaic augmentation and gradient accumulation, which are effective in other YOLO implementations, did not significantly improve detection in our application. Therefore, they were excluded from the final model configuration.

4 Evaluation

We evaluate WoodYOLO on a dataset constructed for automating the detection and identification of vessel elements in hardwood species, a critical step toward wood species classification. Vessel elements are the water-conducting cells in hardwoods, that differ from genus to genus due to their characteristic morphological features. These vessel elements provide vital information for wood identification and are easily to distinguish from other cell types like fibers or parenchyma cells.

In this paper, we are specifically concerned with improving the localization of these vessel elements. The dataset comprises high-resolution microscope images

of macerated hardwood samples, captured with a ZEISS Axioscan 7 microscope. Each image, originally in the czi format with a resolution of approximately 54,000 x 31,000 pixels and file size of 1 GB, was scaled down by 10% (5,400 x 3,100 pixels) to enhance training efficiency and reduce memory usage. The final dataset consists of 767 images annotated with 118,287 bounding boxes identifying vessel elements.

Only the third of five focal planes of each image was utilized for training, as additional planes did not contribute significant information for detecting the vessel elements. The annotated dataset was split into 613 images for training and 154 images for validation. We have conducted initial experiments with 5-fold cross-validations, but found that the metrics are relatively stable across different folds. Due to time constraints, we use a simple train-validation split.

In this section, we evaluate the performance of our vessel detection framework across various configurations and compare it to other state-of-the-art models. The evaluations were conducted using the F2 score at a fixed IoU threshold of 0.3, as described before.

4.1 Detection Model Comparison

Since we use YOLO as a basis, it is useful to compare our model with other YOLO variants. In Tab. 1, we present the F2 scores for different detection models.

Architecture	F2 Score
YOLOv10-S	0.691
YOLOv10-M	0.719
YOLOv7-W6	0.783
YOLOv7-tiny	0.723
Ours	**0.848**

Table 1: Comparison of detection models based on the F2 score. "Ours" refers to our best WoodYOLO configuration. The parameters of YOLOv10 and YOLOv7 have both been optimized. It is worth noting that we use a resolution of 5184x5184 pixels for the second-best model YOLOv7-W6, which requires the use of an A100. Our model uses 2048x2048 and can be trained with less than 10 GB of VRAM.

Our customized YOLO variant outperforms other models, achieving an F2 score of 0.848, highlighting its superior ability to detect vessel elements in large microscopic images.

4.2 Backbone Analysis

We evaluated various backbone networks to determine their impact on detection performance. Table 2 summarizes the results, while including the number of parameters.

Backbone	Parameters (M)	F2 Score
YOLOv7-tiny [41]	9.77	0.8146
VGG11-bn [34]	16.59	**0.8316**
RepVGG-A0 [4]	15.52	0.8168
ResNet-18 [10]	18.48	0.8096
EfficientNet-B0 [35]	10.78	0.8198
ConvNeXt-Nano [19]	22.33	0.8284

Table 2: Comparison of different backbone networks. We used 2 neighbors for this experiment.

The VGG11-bn backbone yielded the highest F2 score (0.8316) while maintaining a reasonable parameter count and VRAM usage. All the other backbones except YOLOv7-tiny have much higher VRAM requirements as they use skip connections, more complex activation functions or special layers like Squeeze-and-Excitation blocks [13]. The simplicity of VGG makes it possible to scale it easier to higher resolutions.

4.3 Effect of Neighboring Cells

We assessed the impact of considering neighboring grid cells (multi-positives) for matching ground truth boxes. As shown in Table 3, using 0 neighboring cells produced the highest F2 score (0.8481).

Number neighbors	F2 Score
0	**0.8481**
2	0.8316
4	0.8080

Table 3: Effect of using neighboring grid cells on detection performance.

Adding more neighboring cells led to a decrease in performance, suggesting that the decrease in precision is too high.

4.4 IoU Loss Function Comparison

We compared different IoU-based loss functions to determine their effectiveness in our model. Table 4 shows that the generalized IoU (GIoU) loss yielded the best performance with a F2 score of 0.8340.

However, the differences at F2 are quite small. This parameter therefore has no major influence on the result.

IoU loss	F2 Score
ciou	0.8316
diou	0.8321
iou	0.8293
giou	**0.8340**

Table 4: Comparison of different IoU-based loss functions. We used 2 neighbors for this experiment.

4.5 Impact of Image Size

Table 5 evaluates the impact of varying image sizes on detection performance. Training on images of size 2048 provided the highest F2 score (0.8316).

Image size	F2 Score
1024	0.7863
2048	**0.8316**
4096	0.8243

Table 5: Effect of different image sizes on detection performance. We used 2 neighbors and ciou for this experiment.

This confirms that we do not need the full resolution of 54000 x 31000 to find the vessel elements. Therefore, it is also not necessary to split the images to perform the detection for individual patches. Since only a single image needs to be predicted with our approach, we have a higher prediction speed.

We have successfully trained a model with a resolution of 6144 x 6144 on an A100 GPU with 40 GB VRAM. Even higher resolutions are possible with further adjustments to the architecture. It is important to emphasize that our standard model, which operates at a resolution of 2048 x 2048, is designed to be more accessible. It can be trained on consumer-grade hardware and requires only about 8 GB of VRAM for training.

4.6 Training Techniques and Anchors

In training our YOLO-based model, we explored several advanced techniques to enhance performance, including mosaic augmentation and gradient accumulation. Mosaic augmentation is a data augmentation strategy that creates a new training image by combining four different images from the dataset. This technique is intended to provide more context and variability during training, potentially improving the model's generalization ability. However, as shown in Tab. 6, mosaic augmentation did not lead to an improvement in the F2 score for our task.

Method	F2 Score
Baseline (Ours)	**0.848**
+ Mosaic Augmentation	0.786
+ Gradient Accumulation	0.838
+ No Maximum Size Constraint	0.841

Table 6: Comparison of approaches to increase F2.

Gradient accumulation is another technique we evaluated. It allows for effective training with larger batch sizes than can fit in GPU memory by accumulating gradients over multiple mini-batches before updating the model weights. Despite its potential to stabilize training and improve convergence, our results indicate that gradient accumulation did not provide a significant benefit in our experiments.

One key modification that proved beneficial was the implementation of a maximum object width and height (the previously discussed anchor box variant). Removing this constraint resulted in a noticeable decrease in the F2 score, demonstrating the effectiveness of this technique in improving detection performance.

4.7 Summary of the results

We have demonstrated that WoodYOLO outperforms other YOLO variants in our specific use case. Interestingly, certain techniques that have consistently shown improvements in mAP on COCO do not yield similar benefits here. For instance, mosaic augmentation, introduced in YOLOv4 [2], showed a 1.8% increase in AP_{50} in their ablation study. In contrast, our experiments reveal a substantial decrease of 6.2% in F2 score when applying this technique. Similarly, we observed no advantage in using multi-positives, despite YOLOX reporting a 2.1% improvement.

We attribute these discrepancies to several factors:

- Metric difference: Our focus is on recall and approximate bounding box overlap, rather than the standard COCO metrics.
- Task simplification: As we only need to localize objects, our architecture can be shallower compared to those designed for more complex tasks.
- Reproducibility challenges: Deep learning, particularly in object detection, often faces reproducibility issues. Many YOLO implementations use legacy code with undocumented workarounds to improve AP, which are not mentioned in the original papers. These may include arbitrary loss function weightings or different weight decay strategies [11].

To mitigate these confounding factors, we developed our detector from scratch, avoiding reliance on previous codebases. This approach allows us to more accurately assess the impact of individual modifications.

In conclusion, our findings suggest that for specialized domains that diverge significantly from the standard COCO use-case, developing customized detectors can be more beneficial than adapting existing general-purpose models. This approach enables a more tailored solution that better addresses the specific requirements of the task at hand.

5 Conclusion

In this paper, we presented WoodYOLO, a novel object detection algorithm specifically designed for microscopic wood fiber analysis. Our approach builds upon the YOLO architecture, incorporating tailored optimizations to enhance performance in high-resolution microscopy images. We introduced several key innovations, including a customized YOLO-based architecture optimized for microscopic images and a novel anchor box specification method.

Our comprehensive evaluation demonstrated that WoodYOLO outperforms state-of-the-art models such as YOLOv10 and YOLOv7 by significant margins in terms of F2 score. We also provided insights into the effectiveness of various architectural decisions and training techniques in the context of wood vessel detection.

The superior performance of WoodYOLO in detecting vessel elements in microscopic images of fibrous materials represents a significant advancement in automated wood species identification. This contribution has far-reaching implications for enhancing regulatory compliance, supporting sustainable forestry practices, and promoting biodiversity conservation efforts globally.

5.1 Future Work

The development of WoodYOLO opens up several promising avenues for future research and improvement. A key area for exploration is the integration of rotated bounding boxes to improve the accuracy of vessel element localization, particularly for elongated or angled structures. This further development requires adjustments to both the model architecture and the dataset annotations and offers considerable potential for improving detection accuracy.

At the same time, further optimization of the WoodYOLO architecture can be worked on to reduce the GPU requirements and increase the recall. Reducing the model's memory requirements is crucial to enable the processing of larger, higher-resolution microscopic images.

In addition to these technical improvements, we see great potential for adapting WoodYOLO to other areas that require high-resolution image analysis. For example, our approach could be useful in medical imaging to detect cell structures or in analyzing satellite imagery to identify specific geographical features. By exploring these cross-domain applications, we aim to extend the impact of our research beyond forestry and wood science and potentially contribute to advances in various scientific disciplines.

References

1. Aldughayfiq, B., Ashfaq, F., Jhanjhi, N.Z., Humayun, M.: Yolov5-fpn: A robust framework for multi-sized cell counting in fluorescence images. Diagnostics **13**(13), 2280 (Jul 2023). https://doi.org/10.3390/diagnostics13132280, http://dx.doi.org/10.3390/diagnostics13132280 4
2. Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M.: Yolov4: Optimal speed and accuracy of object detection (2020), https://arxiv.org/abs/2004.10934 3, 13
3. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers (2020), https://arxiv.org/abs/2005.12872 3
4. Ding, X., Zhang, X., Ma, N., Han, J., Ding, G., Sun, J.: Repvgg: Making vgg-style convnets great again (2021), https://arxiv.org/abs/2101.03697 11
5. Everingham, M., Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The Pascal Visual Object Classes (VOC) Challenge. Int. J. Comput. Vision **88**(2), 303–338 (jun 2010). https://doi.org/10.1007/s11263-009-0275-4, https://doi.org/10.1007/s11263-009-0275-4
6. Feng, C., Zhong, Y., Gao, Y., Scott, M.R., Huang, W.: Tood: Task-aligned one-stage object detection (2021), https://arxiv.org/abs/2108.07755 9
7. Flaig, M.L., Berger, J., Wenig, P., Olbrich, A., Saake, B.: Identification of tropical wood species in paper: a new chemotaxonomic method based on extractives. Holzforschung **77**(11-12), 860–878 (2023). https://doi.org/doi:10.1515/hf-2023-0048, https://doi.org/10.1515/hf-2023-0048 1
8. Ge, Z., Liu, S., Li, Z., Yoshie, O., Sun, J.: Ota: Optimal transport assignment for object detection (2021), https://arxiv.org/abs/2103.14259 9
9. Ge, Z., Liu, S., Wang, F., Li, Z., Sun, J.: Yolox: Exceeding yolo series in 2021 (2021), https://arxiv.org/abs/2107.08430 3, 8, 9
10. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition (2015), https://arxiv.org/abs/1512.03385 4, 11
11. He, T., Zhang, Z., Zhang, H., Zhang, Z., Xie, J., Li, M.: Bag of tricks for image classification with convolutional neural networks (2018), https://arxiv.org/abs/1812.01187 13
12. Helmling, S., Olbrich, A., Heinz, I., Koch, G.: Atlas of vessel elements: Identification of asian timbers. Iawa Journal **39**(3), 249–352 (2018) 1
13. Hu, J., Shen, L., Albanie, S., Sun, G., Wu, E.: Squeeze-and-excitation networks (2019), https://arxiv.org/abs/1709.01507 11
14. Huang, X., Wang, X., Lv, W., Bai, X., Long, X., Deng, K., Dang, Q., Han, S., Liu, Q., Hu, X., Yu, D., Ma, Y., Yoshie, O.: Pp-yolov2: A practical object detector (2021), https://arxiv.org/abs/2104.10419 3
15. Ilvessalo-Pfäffli, M.S.: Fiber atlas: identification of papermaking fibers. Springer Science & Business Media (1995) 1
16. Li, C., Li, L., Jiang, H., Weng, K., Geng, Y., Li, L., Ke, Z., Li, Q., Cheng, M., Nie, W., Li, Y., Zhang, B., Liang, Y., Zhou, L., Xu, X., Chu, X., Wei, X., Wei, X.: Yolov6: A single-stage object detection framework for industrial applications (2022), https://arxiv.org/abs/2209.02976 3
17. Li, P., Che, C.: Semo-yolo: A multiscale object detection network in satellite remote sensing images. In: 2021 International Joint Conference on Neural Networks (IJCNN). pp. 1–8 (2021). https://doi.org/10.1109/IJCNN52387.2021.9534343 4

18. Lin, T.Y., Maire, M., Belongie, S., Bourdev, L., Girshick, R., Hays, J., Perona, P., Ramanan, D., Zitnick, C.L., Dollár, P.: Microsoft coco: Common objects in context (2015), https://arxiv.org/abs/1405.0312 3
19. Liu, Z., Mao, H., Wu, C.Y., Feichtenhofer, C., Darrell, T., Xie, S.: A convnet for the 2020s (2022), https://arxiv.org/abs/2201.03545 4, 11
20. Long, X., Deng, K., Wang, G., Zhang, Y., Dang, Q., Gao, Y., Shen, H., Ren, J., Han, S., Ding, E., Wen, S.: Pp-yolo: An effective and efficient implementation of object detector (2020), https://arxiv.org/abs/2007.12099 3
21. López Flórez, S., González-Briones, A., Hernández, G., Ramos, C., de la Prieta, F.: Automatic cell counting with yolov5: A fluorescence microscopy approach. International Journal of Interactive Multimedia and Artificial Intelligence **8**(3), 64 (2023). https://doi.org/10.9781/ijimai.2023.08.001, http://dx.doi.org/10.9781/ijimai.2023.08.001 4
22. Meng, X., Li, C., Li, J., Li, X., Guo, F., Xiao, Z.: Yolov7-ma: Improved yolov7-based wheat head detection and counting. Remote Sensing **15**(15) (2023). https://doi.org/10.3390/rs15153770, https://www.mdpi.com/2072-4292/15/15/3770 4
23. Nieradzik, L., Sieburg-Rockel, J., Helmling, S., Keuper, J., Weibel, T., Olbrich, A., Stephani, H.: Automating wood species detection and classification in microscopic images of fibrous materials with deep learning (2023) 2, 3
24. Ouyang-Zhang, J., Cho, J.H., Zhou, X., Krähenbühl, P.: Nms strikes back (2022), https://arxiv.org/abs/2212.06137 3
25. Parliament, E.: Regulation (eu) 2023/1115 of the european parliament and of the council of 31 may 2023 on the making available on the union market and the export from the union of certain commodities and products associated with deforestation and forest degradation and repealing regulation (eu) no 995/2010. Off. J. Eur. Union **150**, 206–247 (2023) 1
26. Ravindran, P., Thompson, B.J., Soares, R.K., Wiedenhoeft, A.C.: The xylotron: flexible, open-source, image-based macroscopic field identification of wood products. Frontiers in plant science **11**, 1015 (2020) 2
27. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection (2016), https://arxiv.org/abs/1506.02640 3
28. Redmon, J., Farhadi, A.: Yolo9000: Better, faster, stronger (2016), https://arxiv.org/abs/1612.08242 3, 6, 9
29. Redmon, J., Farhadi, A.: Yolov3: An incremental improvement (2018), https://arxiv.org/abs/1804.02767 3
30. Rezatofighi, H., Tsoi, N., Gwak, J., Sadeghian, A., Reid, I., Savarese, S.: Generalized intersection over union: A metric and a loss for bounding box regression (2019), https://arxiv.org/abs/1902.09630 7
31. Ruffinatto, F., Crivellaro, A.: Atlas of macroscopic wood identification: with a special focus on timbers used in Europe and CITES-listed species. Springer Nature (2019) 1
32. Schmitz, N., Beeckman, H., Blanc-Jolivet, C., Boeschoten, L., Braga, J.W., Cabezas, J.A., Chaix, G., Crameri, S., Deklerck, V., Degen, B., et al.: Overview of current practices in data analysis for wood identification. a guide for the different timber tracking methods. Tech. rep. (2020) 1
33. Silva, J.L., Bordalo, R., Pissarra, J., de Palacios, P.: Computer vision-based wood identification: A review. Forests **13**(12), 2041 (2022) 2
34. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition (2015), https://arxiv.org/abs/1409.1556 4, 11
35. Tan, M., Le, Q.V.: Efficientnet: Rethinking model scaling for convolutional neural networks (2020), https://arxiv.org/abs/1905.11946 11

36. Tian, Z., Shen, C., Chen, H., He, T.: Fcos: Fully convolutional one-stage object detection (2019), https://arxiv.org/abs/1904.01355 8
37. Tsuchikawa, S., Kobori, H.: A review of recent application of near infrared spectroscopy to wood science and technology. Journal of Wood Science **61**(3), 213–220 (2015) 1
38. UTAR, FRIM: Mywood-premium (2018), https://mywoodid.frim.gov.my/, (accessed on 15 May 2023) 2
39. Wang, A., Chen, H., Liu, L., Chen, K., Lin, Z., Han, J., Ding, G.: Yolov10: Real-time end-to-end object detection (2024), https://arxiv.org/abs/2405.14458 3
40. Wang, C.Y., Bochkovskiy, A., Liao, H.Y.M.: Scaled-yolov4: Scaling cross stage partial network (2021), https://arxiv.org/abs/2011.08036 3
41. Wang, C.Y., Bochkovskiy, A., Liao, H.Y.M.: Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors (2022), https://arxiv.org/abs/2207.02696 3, 5, 9, 11
42. Wang, C.Y., Liao, H.Y.M., Yeh, I.H., Wu, Y.H., Chen, P.Y., Hsieh, J.W.: Cspnet: A new backbone that can enhance learning capability of cnn (2019), https://arxiv.org/abs/1911.11929 4
43. Wang, C.Y., Yeh, I.H., Liao, H.Y.M.: Yolov9: Learning what you want to learn using programmable gradient information (2024), https://arxiv.org/abs/2402.13616 3
44. Wiedenhoeft, A.C.: The xylophone: toward democratizing access to high-quality macroscopic imaging for wood and other substrates. Iawa Journal **41**(4), 699–719 (2020) 2
45. Xu, S., Wang, X., Lv, W., Chang, Q., Cui, C., Deng, K., Wang, G., Dang, Q., Wei, S., Du, Y., Lai, B.: Pp-yoloe: An evolved version of yolo (2022), https://arxiv.org/abs/2203.16250 3
46. Xu, X., Jiang, Y., Chen, W., Huang, Y., Zhang, Y., Sun, X.: Damo-yolo : A report on real-time object detection design (2023), https://arxiv.org/abs/2211.15444 3
47. Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., Ni, L.M., Shum, H.Y.: Dino: Detr with improved denoising anchor boxes for end-to-end object detection (2022), https://arxiv.org/abs/2203.03605 3
48. Zhao, Y., Lv, W., Xu, S., Wei, J., Wang, G., Dang, Q., Liu, Y., Chen, J.: Detrs beat yolos on real-time object detection (2024), https://arxiv.org/abs/2304.08069 3
49. Zheng, Z., Wang, P., Liu, W., Li, J., Ye, R., Ren, D.: Distance-iou loss: Faster and better learning for bounding box regression (2019), https://arxiv.org/abs/1911.08287 7
50. Zheng, Z., Wang, P., Ren, D., Liu, W., Ye, R., Hu, Q., Zuo, W.: Enhancing geometric factors in model learning and inference for object detection and instance segmentation (2021), https://arxiv.org/abs/2005.03572 7