Automatic detection of ~~floating~~ floating instream large wood in videos using deep learning

Janbert Aarnink[1], Tom Beucler[1,2], Marceline Vuaridel[1], and Virginia Ruiz-Villanueva[1,3]

[1]Université de Lausanne, Faculty of Geosciences and the Environment (FGSE), Institute of Earth Surface Dynamics (IDYST), Quartier UNIL-Mouline - Bâtiment Géopolis, 1015 Lausanne, Switzerland
[2]Université de Lausanne, Expertise Center for Climate Extremes, 1015 Lausanne, Switzerland
[3]University of Bern, Institute of Geography, Hallerstrasse 12, 3012, Bern, Switzerland

Correspondence: Janbert Aarnink (janbert.aarnink@unil.ch)

Abstract. Instream large wood (i.e., downed trees, branches, and roots larger than 1m in length and 10cm in diameter) has essential ~~geopmorphological~~ geomorphological and ecological functions supporting the health of river ecosystems. ~~Still~~However, even though its transport during floods may pose a risk, it is rarely observed and ~~, therefore,~~ poorly understood. This paper presents a novel approach to detect floating pieces of instream wood from ~~video~~videos. The approach uses a Convolutional Neural Network to detect wood automatically. We sampled data to represent different wood transport conditions, combining 20 datasets to yield thousands of instream wood images. We designed multiple scenarios using different data subsets with and without data augmentation~~and analyzed~~. We analysed the contribution of each one to the effectiveness of the model using k-fold cross-validation. The mean average precision of the model varies between 35~~%~~ and 93~~percent, and is highly~~ %, which is influenced by the quality of the data ~~which~~ it detects. When ~~the image resolutionis low, the identified components in the labeled pieces, rather than exhibiting distinct characteristics such as bark or branches, appear more akin to amorphous masses or 'blobs'. We found that~~ using a 418-pixel input image resolution, the model detects wood with ~~a~~ an overall mean average precision of 67~~percent when using a 418 pixels input image resolution. Also, improvements~~%. Improvements of up to 23~~percent % mean average precision~~ could be achieved in some instances, and increasing the input resolution raised the weighted mean average precision to 74~~percent. We show~~ %. We demonstrate that the detection performance on a specific dataset is not solely determined by the complexity of the network or the training data. Therefore, the findings of this paper can be used when designing a custom wood detection network. With the growing availability of flood-related videos featuring wood uploaded to the ~~internet~~Internet, this methodology facilitates the quantification of wood transport across a wide variety of data sources.

1 Introduction

Instream large wood includes downed trees, root wads, trunks, and branches of at least 10 centimetres in diameter and 1 ~~meter~~ metre in length (Platts et al., 1987)~~and~~. It is typically recruited from forested areas within the river catchment by natural tree mortality, wind storms, snow avalanches, wildfires, landslides, debris flows~~and bank erosion . As it distributes along the river banks, wood~~, bank erosion as well as beaver activity (Benda and Sias, 2003). Stored wood within the river corridor plays a crucial role by trapping sediment, creating pools, and generating spatially varying flow patterns (Keller et al., 1995; Andreoli

et al., 2007; Wohl et al., 2018). Therefore, instream wood is a crucial driver of the rivers' form and functioning and positively influences the diversity of the river ecosystem (Wohl et al., 2017). ~~Whilst~~ Although beneficial for biodiversity, wood can also be a hazard. ~~Given that wood is typically transported during high flow conditions, floods often significantly influence the redistribution of instream wood.~~ During floods, large quantities of transported wood may accumulate at bridges or narrow river sections, blocking the channel and causing ~~larger~~ localised inundations (Lucía et al., 2015). Additionally, the ~~pressure from wood can become large enough to damage or even wipe out complete bridges (Diehl, 1997; Lyn et al., 2003)~~ accumulation of wood can damage or eventually destroy bridges (Diehl, 1997; Lyn et al., 2003; De Cicco et al., 2018; Pucci et al., 2023). Costly wood removal efforts have long been the default mitigation strategy (Wohl, 2014), without considering the ecomorphological impact (Lassettre and Kondolf, 2012; Collins et al., 2012). However, these preventive efforts can even be counterproductive. ~~Logjams~~ For example, natural wood accumulations upstream from infrastructure can trap more wood transported during high-flow events, preventing it from accumulating at ~~critical infrastructures downstream~~ the critical infrastructure downstream (Ruiz-Villanueva et al., 2017). A more complex river system resulting from instream wood can also dissipate more flood energy ~~when compared to a channelized~~ than a channelised river (Curran and Wohl, 2003; Hassan et al., 2005). ~~By~~ Human influence has impacted wood regimes and the river ecosystem by building infrastructure, channelling, and removing wood from rivers~~, human influence has impacted wood regimes and the river ecosystem. To assess the health of a river~~ (Wohl et al., 2019). Therefore, it is crucial to ~~get a better understanding of~~ understand instream large wood dynamics by assessing the quantity ~~and transport.The amount of observations of instream wood is, however, scarce.To improve our knowledge of wood dynamics in various rivers, we can analyze monitoring systems or~~ (i.e., wood supply and storage) and transport or fluxes. In addition to direct monitoring and crowdsourced videos of floods~~to detect and quantify the flow of instream wood . While existing detection algorithms have proven effective at locations to which they are calibrated, they lack widespread applicability, limiting mitigation strategies.~~

~~Wood transport data is scarce. Although rarely monitored, estimating~~, recent advancements in understanding large wood dynamics have emerged through experimental studies in flumes and numerical studies (Pancini, 2021; Innocenti et al., 2023). Estimating the quantity of wood in river systems and its temporal variation has gained traction over the last ~~years.~~ few years. However, in-field wood transport data remains scarce. As wood is mainly transported during floods, observations of transported instream wood are rare, and very few rivers are currently being monitored with that aim (Ghaffarian et al., 2020, 2021). Different techniques can help assess a river's wood regime in terms of transport, such as Radio Frequency Identification (RFID), high-resolution aerial surveys, and video monitoring (MacVicar et al., 2009). With RFID tags, individual pieces of wood are given a unique identity, and their movement can be registered and tracked. RFID tags can be used to quantify the percentage of wood that moves each year (Schenk et al., 2013). Attaching GPS loggers to pieces of instream wood is expensive and limited in temporal range, but it can give temporal data with a high frequency during high discharge events (Ravazzolo et al., 2015). Aerial data can detect stored wood and wood jams (Haschenburger and Rice, 2004; Lassettre et al., 2008; Sanhueza et al., 2018). However, the best methods to quantify wood transport are video-based because such methods provide high temporal and spatial resolution (Ghaffarian et al., 2020). Before introducing deep learning methods, conventional computer vision methods were used for object detection. These methods rely on feature extraction techniques such as edge detection, background subtraction,

template matching or Histogram of Oriented Gradients (HOG) (Zou, 2019). Edge detectors use pixel-based filters to analyse

60   changes in image intensity (Sun et al., 2022). They help detect the contours of an object. Background subtraction algorithms work well with static camera setups (Kalsotra and Arora, 2021). It models the background and subtracts the background model from the current frame. Template matching techniques involve overlaying a template image across the input image to find regions that match the template (Swaroop and Sharma, 2016). Much like edge detectors, the HOG extracts features by counting the occurrences of gradient orientation in certain portions of the image (Dalal and Triggs, 2005). Although robust, it requires

65   careful tuning.

Using computer vision software combined with stationary cameras for detecting wood transport has provided a first insight into river wood dynamics (Lemaire et al., 2015; Zhang et al., 2021). The approach uses spatial and temporal pixel-level analyses to, for example, detect features like colours, edges, and moving objects. Its first feature is a mask to identify potential floating objects that differ in colour from the water's surface. Combining these features eventually ~~determines the likelyhood of wood~~

70   ~~being present~~allows to ascertain the presence of wood in the images. Even though the ~~utility of the software was proven , and~~ ~~it~~ approach's utility was proven and is used to extract wood from videos at ~~multiple~~ a few sites (Zhang et al., 2021), it still requires manual, site-specific tuning. It is purposefully designed for a specific site to increase performance. When creating a method for a specific location, it performs well on data with which it is designed but becomes too specific and complex to generalize over a wide variety of datasets. Furthermore, the current method requires the camera to be angled in a fixed position

75   to extract the wood-detection features, which decreases flexibility. Even when tuned to a specific location, its performance depends on seasonal and weather conditions (Ghaffarian et al., 2021). Furthermore, measuring stations are limited by their spatial locations and rely on specific installation setups before a wood-moving event.

Developments in mobile technology have enabled millions of people to use high-quality video cameras. During extreme weather events, videos of floods are often posted online, which can be an exciting source for wood transport ~~analyzes~~analyses.

80   Citizen science projects like the Argentinian Storm Chasers project have ~~shown~~ demonstrated the use of home videos to ~~analyze~~ analyse hydraulic conditions during a storm (Le Coz et al., 2016). Similarly, crowdsourced videos can ~~analyze~~ analyse wood regime characteristics (Ruiz-Villanueva et al., 2019). However, quantifying wood transport from videos recorded from non-fixed standpoints presents a challenge, as existing tracking methods fail to ~~analyze~~ analyse crowdsourced footage due to their dependency on a stationary camera angle~~, necessitating a more flexible wood detection solution.~~. Manual detection and

85   quantification of large wood have been conducted in previous studies (Ruiz-Villanueva et al., 2022); however, this process is labour-intensive. Advances in machine learning methods can be applied flexibly and could allow widespread wood detection.

~~After first being proposed in the 1940s (Prieto et al., 2016), the neural network has made a recent come-back that went hand~~ ~~in hand with developing high-performance parallel computing systems (Zhao et al., 2019). The neural network has recently~~ ~~been found to be~~ Convolutional Neural Networks (CNNs) have been identified as an effective method for object detection

90   (Lecun et al., 2015), ~~and was used for instream plastic detection (van Lieshout et al., 2020). In our~~ showing success in remote sensing environmental monitoring applications (Li et al., 2020). These methods have been utilized for various tasks, including segmenting tree trunks in urban areas (Jodas et al., 2021), detecting floating plastic debris in rivers (van Lieshout et al., 2020; Àlex Solé Gó , and monitoring river flow (Dibike and Solomatine, 2001). In the studies that highlighted the potential of deep learning for

detecting and classifying objects in fluvial environments, transfer learning was used to classify static stored large wood in rivers from aerial imagery, achieving a recall rate of 93.75%, to overcome data scarcity (Schwindt et al., 2024). Similarly, deep learning has proven effective in other aquatic contexts, such as fish species detection and weight estimation (Sokolova et al., 2023), further illustrating the adaptability of CNNs in riverine environments. However, the application of CNNs for automating floating wood detection remains under-explored, which we attribute to the lack of uniform training data. In this study, we propose using a ~~Convolutional Neural Network (CNN) to provide~~ YOLO CNN as a flexible first ~~step to analyze wood transport~~ approach to analyse floating wood in videos from various sources ~~and circumvent~~, circumventing the limitations of ~~the~~ current state-of-the-art wood detection ~~algorithm. The CNN has~~ algorithms, which are site-specific and require calibration. Our algorithm aims to detect and track floating wood pieces in any river under various conditions and from varying sources. Different video sources could include permanent monitoring cameras and handheld devices from witnesses of wood-laden flood events in rivers. This offers immediate applications such as computing wood fluxes to understand wood dynamics in rivers and practical uses like warning systems, flood hazard and risk assessments.

2 Methods

2.1 Convolutional neural network selection

Our Convolutional Neural Network (CNN) comprises multiple convolutional layers ~~in which it analyzes~~ that analyse video frames. Convolutions are used to extract hierarchical features from images to make predictions ~~. Features like~~ (Lecun et al., 2015). Features such as edges, corners, and textures are combined to determine the class of an object. The algorithm ~~can learn which features an object needs to have for it to be classified as a piece of~~ learns which features are necessary for classifying an object as instream wood. These detection features do not ~~have to be individually hard-coded and are created~~ require individual hard-coding but are developed by training the network with class examples. Depending on the architecture of the CNN, it can ~~therefore be a couple of orders of magnitudes~~ thus be several orders of magnitude more complex and theoretically more effective at detecting wood. ~~The training of a CNN requires a large amount of~~ Training a CNN demands substantial data, ideally from ~~vastly different sources in different~~ diverse sources under varying weather and flow conditions (Bengio et al., 2013). ~~Popular CNN architectures that have been developed for object detection include the Single Shot MultiBox~~

The main Convolutional Neural Networks (CNNs) using deep learning include the Region-Based Convolutional Neural Network (R-CNN), the Single Shot Multibox Detector (SSD), ~~the faster R-CNN, the CenterNet, Retina-Net and~~ CenterNet, and the You Only Look Once (YOLO) ~~(Liu et al., 2016; Ren et al., 2017; Duan et al., 2019; Lin et al., 2020; Bochkovskiy et al., 2020)~~.

3 ~~Methods~~

~~For~~ algorithm. The R-CNN introduced the concept of region proposal networks (RPN) that first extract a region of interest before classifying it (Ren et al., 2017). This two-step process generally results in longer processing times. The SSD method, in

125 contrast, does not use the region proposal step but instead utilises multi-scale feature maps (Liu et al., 2016). Another approach that omits the proposal stage is CenterNet, which identifies an object as a pair of key points representing the centre and size of the object (Duan et al., 2019). The YOLO algorithm features a unified architecture and detects objects as a single regression task, from image pixels to bounding box coordinates and class probabilities. This enables YOLO to excel in speed and accuracy (Bochkovskiy et al., 2020). Its single-pass architecture allows it to consider the entire image context during the detection phase,

130 which is advantageous for identifying multiple pieces of wood simultaneously (Redmon et al., 2016). Furthermore, the YOLO algorithm has undergone seven major updates over the past six years (Redmon et al., 2016; Wang et al., 2022). These frequent updates and improvements can benefit the development of a wood detection method. Therefore, for the main part of this study, we have chosen to train the fourth generation of the You Only Look Once (YOLO) network. ~~Among other object detection networks, YOLO stands out as it offers both speed and accuracy as it directly predicts bounding boxes and class probabilities~~

135 ~~automatically and thereby outperforms its competitors (Bochkovskiy et al., 2020).~~

2.1 Data

Training a Convolutional Neural Network (CNN) that can detect wood in various conditions requires multiple steps. First, instream wood data is acquired and labelled. Subsequently, the dataset is trimmed and augmented to create a database of varying images containing instream wood. Once these steps are complete, a large part of the database is used to train the

140 model, whilst a ~~small~~ smaller part is used to validate the training performance. In this context, "database" refers to all the data used to create the CNN, whilst a "dataset" is a subset of the database consisting of all the data recorded by the same device at a specific location and date. Figure 1 gives an overview of the data collection and processing ~~. It shows~~ and demonstrates that we assess the performance of 14 different augmentation and sampling strategies from the datasets.
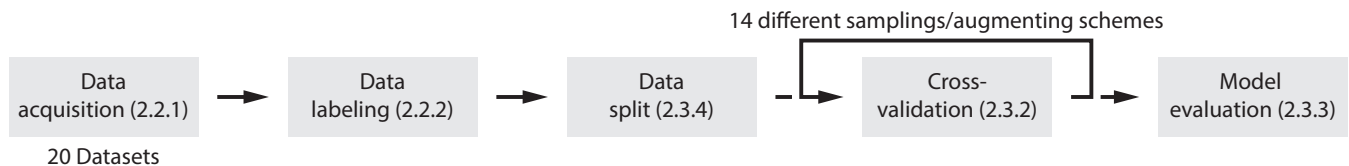
Figure 1. Overview of the methodology used for data collection and processing. Of the 20 labelled datasets obtained, ~~6~~ six representative datasets are chosen for cross-validation (see Figure 4).

2.2 ~~Data~~

145 ### 2.1.1 Data acquisition

For this study, we ~~used~~ employed five low-cost cameras, including three ~~Android phones~~ Android phones and two Raspberry Pi camera modules. The cameras are ~~installed at different~~ temporarily installed at various locations, on different days and times~~of the day, with different orientations~~, with various orientations and resolutions (see table 1). We manually ~~added wood~~

~~to the river~~ introduced wood upstream from the cameras into the river and let them record the wood passing by during a time window ranging from 30 to 90 minutes. We also used data from two locations in France that have been actively monitored ~~for the last years~~ by a permanent camera at the Allier since 2019 and the Ain Rivers ~~(Zhang et al., 2021), and added~~ since 2007 (Zhang et al., 2021; Hortobágyi et al., 2024). This data was used to analyse the wood flux and only contains natural instream wood occurrences. An operator-based visual floating wood detection method was employed to detect the wood. Labels were already available; however, as the labelling process involved only labelling the new pieces in each frame, the labels were insufficient for the method proposed in this paper. Lastly, carefully selected images of floating wood were added from online sources (purchased from istockphoto.com and dreamstime.com) to represent a small but diverse floating wood dataset. All images in this dataset are from different sources and in various locations. Additionally, instead of being recorded from a camera attached to a bridge, these images were captured by photographers. When this dataset is used as training data, it helps increase the variety of data sources. When used as validation data, it is a benchmark on how well the model generalizes wood. The final database consists of 15,228 images, each containing one or more pieces of wood. The images are separated into 20 different datasets. ~~For example, figure ?? shows 9 of the 20 different datasets~~, of which nine are shown in Figure 2.

2.1.2 ~~Labeling~~Data labelling

Labels are created on the data ~~that is~~ acquired to indicate where an instream piece of wood is located within each frame. Bounding boxes represent the four coordinates of a box's corners that fit around the piece of wood (see ~~figure 8~~ Figure 2 for examples of bounding boxes). Initially, the labelling is manually ~~done~~ performed using labelling software called LabelImg (Viso.ai, 2022). To ~~speed up~~ expedite the process, we devised a ~~pseudo-labeling~~ pseudo-labelling method. Only 10~~per cent~~% of the images in each dataset (1922 in total) were labelled by hand. A CNN (CenterNet, (Duan et al., 2019)) was trained to ~~purposefully~~ deliberately overfit that specific dataset by using only images from that particular dataset. With the CNN, the labels for the other 90~~per cent~~% of images are created. ~~After that, the rest of the labels were checked manually. In~~ Subsequently, we verified that all bounding boxes correctly indicated a piece of wood and adjusted the incorrect labels by going through all labels manually. It was verified that this method worked well in 11 out of 15 cases ~~, this method worked well~~and required minimal manual intervention. However, for the other four datasets, the CenterNet's performance was not sufficient to aid in the labelling of the other 90~~per cent~~% of images, as the mean Average Precision (mAP, see section 2.2.3) was below 20~~per cent~~%. Therefore, ~~in these four cases~~it would have required too much time and effort to manually create a completely labelled dataset; therefore, only the ~~hand-labeled~~ hand-labelled 10~~per cent~~% of images were used.

2.1.3 Data analysis

After labelling, we obtained 15,228 fully labelled video screenshots with bounding boxes (see ~~figure ??) around the~~ Figure 2) around 33,160 pieces of instream wood. When training a CNN, the goal is to have a lot of diverse data. ~~Due to the natural way wood drifts and gets stuck~~Naturally, the wood floats and moves in a flow or is deposited or trapped by an obstacle (i.e., river bank, boulders, trees), some videos contain minutes of the same piece recorded at the exact location. Therefore~~the data~~

(a)

(b)

Figure 2. ~~Data acquired at~~ Examples from 9 ~~out~~ of ~~the~~ 20 ~~locations,~~ datasets. (a) Examples with bounding boxes around instream wood. (b) Locations of taken datasets. Images 1-2: La Chamberonne~~at UNIL campus, Switzerland (46.52373N, 6.57729E; 46.52296N, 6.57577E)~~. Image 3: La Borgne d'Arolla~~, Switzerland (46.04814N, 7.48884E)~~. Image 4: Dixence~~, Val d'Herens, Switzerland (46.17966N, 7.4187E)~~. Images 5-6: La Borgne~~, Val d'Herens, Switzerland (46.1612N, 7.44079E; 46.10975N, 7.49428E)~~. Image 7: Ain River~~, France~~. Image 8: Allier River~~, France~~. Image 9~~is included to enhance data diversity~~: unknown location.

~~is~~, the data was trimmed. If, for subsequent frames, the labels were almost identical (with all bounding boxes within a certain percentage of each other), only 1 of the frames was kept in the database.

From the database, all labels were cropped out of their corresponding images and resized to ~~80 by 80 greyscale images, creating~~ grayscale images of a certain resolution. The pixel resolution was maximised to 80×80 pixels, the largest size in which all labels could be processed with the available random access memory. This yielded images of all 33,~~173~~160 pieces of wood in the database without their surrounding image. Subsequently, the images were normalised and centred to eliminate

Table 1. Data acquisition statistics. For datasets 1-15, the devices were temporarily attached to bridges. Datasets 16-19 are taken from permanent monitoring stations.

Dataset number	Amount of ~~labeled~~ images	Amount of unique labels	Device	Resolution	Location ~~dataset (figure ??)~~	Number in ~~example~~ Figure 2
1	~~1.429~~ 1,429	3,743	Raspberry Pi Camera	1920x1440	La Sorge, loc 1	1
2	601	1,598	Raspberry Pi Camera	1920x1440	La Sorge, loc 1	
~~3~~ 3*	~~1.076~~ 1,076	2,930	Samsung Galaxy A4	3264x2448	La Sorge, loc 1	
4	478	1,195	Xiaomi Redmi 4X	4160x3120	La Sorge, loc 1	
5	344	674	Xiaomi Redmi 4X	4160x3120	La Sorge, loc 2	2
6	~~2.478~~ 2,478	5,436	Raspberry Pi Camera	1920x1440	La Sorge, loc 2	
~~7~~ 7*	~~2.146~~ 2,146	4,029	Raspberry Pi Camera	1920x1440	La Sorge, loc 2	
8	191	343	Samsung Galaxy A4	3264x2448	La Sorge, loc 2	
9	18	28	Xiaomi Redmi 2	3328x2496	Borgne d'Arolla	
10	138	256	Raspberry Pi Camera	1920x1440	Borgne d'Arolla	
~~11~~ 11*	~~1.046~~ 1,046	2,116	Raspberry Pi Camera	1920x1440	Borgne d'Arolla	3
12	~~1.034~~ 1,034	1,946	Raspberry Pi Camera	1920x1440	Dixence	4
~~13~~ 13*	157	180	Raspberry Pi Camera	1920x1440	La Borgne	5
14	~~2.340~~ 2,340	4,472	Raspberry Pi Camera	1920x1440	La Borgne	6
15	~~1.236~~ 1,236	2,232	Samsung Galaxy A4	3264x2448	La Borgne	
16	116	~~unknown~~ 229	HDTV720P	640x480	Ain	7
17	81	~~unknown~~ 152	HDTV720P	640x480	Ain	
~~18~~ 18*	176	~~unknown~~ 1,239	Hikvision DS-2CD2T42WD-I8	1920x654	Allier	
19	134	~~unknown~~ 353	Hikvision DS-2CD2T42WD-I8	1920x1080	Allier	8
~~20~~ 20*	9	~~differing~~ 9	~~differing~~ Differing	~~differing~~ Differing	Differing	9
Average	678	1,658		2247x1673		
Total	15,228	33,160				

*Used as representative dataset.

circumstantial and camera-specific white-balance differences. This means that the average pixel intensity of each picture was set to 128 and the maximum or minimum pixel value to 255 or 0, respectively (see Figure 3). ~~We performed a Principal Component Analysis (PCA) on the set using the "clustimage" Python package (Taskesen, 2021). PCA is an unsupervised dimensionality reduction technique and~~ To analyze the variance in the data and perform clustering in an unsupervised manner, we used a Python package called "clustimage" (Taskesen, 2021) along with custom scripts. The analysis consisted of dimensionality reduction and clustering. First, a Principal Component Analysis (PCA) was applied to reduce the dimensionality of the data. The high-dimensional data (80×800 pixels) was transformed into a lower-dimensional space while retaining 98% of the original variance. This reduced the complexity of the dataset, making it more manageable for subsequent clustering. Following

195 the PCA, we clustered the reduced dataset using a k-means algorithm. For four different instances with 2, ~~in our case, clusters of similar images in a predetermined number of clusters. The silhouette score evaluates the similarity of each of the 6400 pixels (arranged as 80 × 80 pixels) within a cluster~~ 4, 6 and 8 clusters, the silhouette score was calculated to evaluate the similarity within the clusters compared to their dissimilarity across ~~different~~ various clusters, effectively gauging the compactness and separation of the clusters.

(a)   (b)
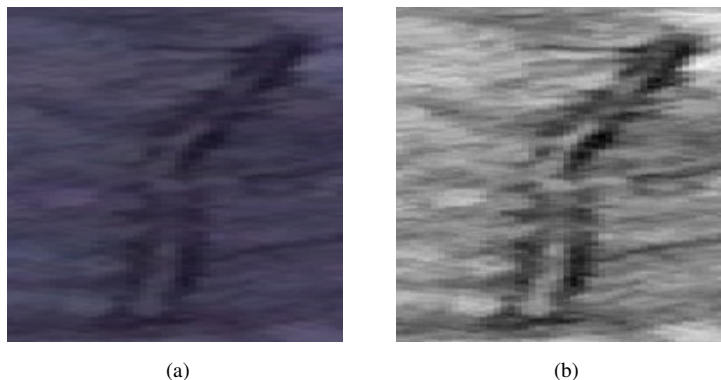
Figure 3. ~~Original cutout~~ (a) Original cutout and (b) greyscaled, ~~normalized~~ normalised, and ~~centered~~ centred cutout ~~(b)~~ (2020-11-29 Raspberry Pi 4 image 7411 label 2).

200 After the ~~PCA~~Principal Components Analysis, we performed a t-Distributed Stochastic Neighbor Embedding (t-SNE)~~to visually review the diversity of cropped-out pieces of labeled wood~~. The stochastic nature of the t-SNE method means that ~~each run might yield different results, so~~ although each run seemed to cluster similar samples, the exact output graph differs each time. Therefore, we used it for ~~visualization purposes only~~~~in our study~~visual interpretation purposes only. Furthermore, we compared the relative size of the bounding boxes ~~from dataset to dataset~~ across datasets to understand the difference in the
205 data.

2.1.4 Data ~~clustering: six representative datasets~~split

~~In machine learning , the data must be split up~~ Machine learning data is typically split into training, validation ~~,~~ and test data~~for evaluation purposes~~. The training data ~~is used to optimize~~ trains the neural network to ~~detect the training examples, the~~ recognise patterns and make predictions. The validation data is used to ~~optimize the neural network~~tune the model's hyperpa-
210 rameters ~~(e.g. , its architecture), and an unseen test dataset is used to analyze the network's performance (Xu and Goodacre, 2018) . Even though the amount of labelled bounding boxes was large, the difference in the data sources was small as the 20 datasets had~~ to ensure it performs well on out-of-sample data. The test data is unseen data used to benchmark the final performance of the CNN (Xu and Goodacre, 2018). Usually, all labelled data is combined, after which a certain portion, such as 90%, is randomly assigned as training data, whilst the other 10% is assigned as validation data. This ensures the training and validation
215 data represent the overall data.

In our case, however, as the data comes from a limited amount of sources with common locations and camera angles. Therefore, the traditional method of , splitting the data into unique train and validation datasets was prone to high variance. To overcome this issue, cross-validation data using traditional methods might cause overfitting, resulting in overestimating the model's performance. A model that overfits performs well on training data but poorly on unseen data because it has learned the specifics of the training data too well. Multiple leave-one-out cross-validations can be used where the same validation technique is used multiple times with different training and to mitigate overfitting, where one complete dataset is left out of the training data and used as validation data.

As the database was separated into 20 different datasets (each with a different camera or location), in an ideal evaluation approach, a leave-one-out cross-validation using 20 different validation datasets would suggest how well different sampling and augmentation scenarios would increase the model 's effectiveness under different circumstances. However, as training 20 models 14 times uses too many resources, the 20 datasets were reduced to 6 representative validation datasets. We chose these six datasets not to be taken on the same day or at the exact location For feasibility purposes, six validation cycles were run. For each cycle, a single dataset was dropped for validation whilst the model was trained on the remaining 19 (see Figure 4). The six representative datasets were chosen to ensure diversity in location, camera angle, and time and reduce computational overhead by avoiding 20 validation cycles. Tests in section 2.2.1 demonstrate that sampling only 500 images from smaller than 500 images were chosen as representative datasets. For each of the six training instances, one of the six representative datasets was left out and used as a validation dataset (see figure 4). We repeated this process training scenario, the performance was averaged over the six runs. This process was repeated for 14 different training scenarios as described in Section section 2.2.2.
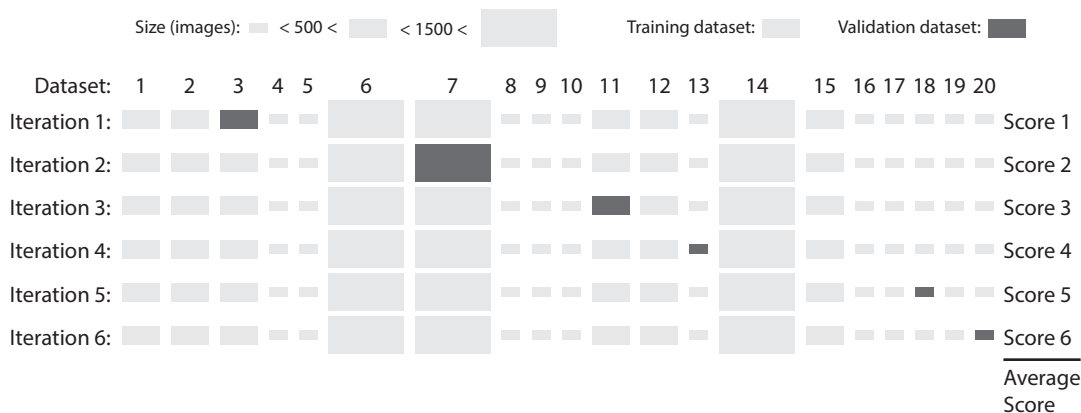
Figure 4. Cross validation scheme for one training scenario. This figure shows the distribution between the datasets used for training and those used for validation. The Y-axis contains all available datasets, and the X-axis contains the different training efforts, in which the . The dark grey dataset represents the validation data whilst , while the other 19 were used for training to train the model. Eventually, the six scores are averaged into the validation score. The size of the rectangle represents the size of the dataset.

Eventually, the scenario that yields the best performance was tested with an independent test dataset. The test dataset consisted of 281 images with a resolution of 1280×720 pixels taken with a Xiaomi Mi 9 phone in timelapse mode (see

figure 7 for an example frame). It was gathered at the river Inn at the moment of an experimental flood at its tributary, the Spöl River. ~~This location has been actively studied~~ The River Ecosystems Research Group at the University of Lausanne has actively studied this location for wood transport since ~~2018 and~~ 2018. It yields a valuable test dataset because an algorithm like ours could ~~reduce human labor significantly~~greatly reduce human labour.

2.2 Machine learning~~methods~~

2.2.1 ~~Sensitivity test~~ Training data size sensitivity

Before attempting to enhance performance across various scenarios, we assessed whether the entire dataset was necessary for model training. It was hypothesized that reducing the training data might not compromise model efficacy while potentially accelerating the training process. The 20 training datasets have an average size of 761 images and a median size of 411, with some datasets containing almost 2,500 images. To reduce computational demands, we conducted tests to evaluate how the number of images sampled per dataset affects training performance~~, aiming to minimize computations for later scenarios~~. We conducted two tests with a fourfold difference in the total number of training images compared. In the first test, 2,000 images were augmented and sampled per dataset. Since the datasets did not all contain this exact number of images, they were randomly ~~up- or down-sampled~~ over or under-sampled to reach this figure. When oversampling, no additional data was introduced; instead, it ensured that dataset sizes were equalised, thereby preventing biases in the rewards towards any particular dataset. We applied the same approach for the second test but with 500 images per dataset. Thus, a total training sample size of 38,000 images for the first test (2,000 images per dataset across 19 training datasets) was compared to 9,500 for the second test (500 images per 19 datasets). Following this comparison, we determined ~~the optimal sample size for further exploration~~whether using less total data would decrease the model's performance.

2.2.2 ~~Thirteen training scenarios~~Cross-validation procedure

A series of experiments were conducted to enhance the model's performance. Including the baseline, fourteen different testing scenarios were performed to test the model's sensitivity to stationary frames, dataset size, data augmentation, and data quality. As CNNs have not yet been trained ~~to detect~~ for detecting instream large wood, thorough testing of different training strategies is crucial. To enhance model performance, the database size can be expanded by adding more labelled images or through synthetic augmentation of existing data. ~~Additionally~~Although more effective with image classification practices, augmenting data for object detection has been known to improve model performance by up to 2.7 percentage points mean Average Precision (mAP) in some cases (Zoph et al., 2020). The augmentation practice, however, attracts less research attention because it is considered to transfer poorly to different datasets. Apart from augmentation, employing various sampling strategies may also improve the algorithm's detection performance. The data used for wood detection can also ~~vary significantly regarding camera angle, pixel size~~exhibit diverse camera angles, pixel sizes, and proximity to the stream (see Table 1). To determine the most effective sampling and augmentation strategies for different data types, 14 models were trained and evaluated against a baseline

model. The baseline model was trained using only the labelled images without any modifications. The other 13 scenarios are detailed below:

1. **Trimmed, testing sensitivity to handling stationary frames**: When labels were similar in at least three subsequent frames, the images were deleted from the database. Determining the exact pixels where a bounding box began and ended was sometimes challenging (e.g. when part of a log was underwater and its end was not clearly visible). As a result, bounding boxes around an immobile object could vary from frame to frame. To account for this, detections were considered similar when all bounding boxes were within 4% of their subsequent x and y locations in the frame and 30% of their width and height. These thresholds were determined manually by testing various percentages on cases where multiple stationary logs were detected in successive frames. In this scenario, 13,375 images from the total database were kept.

2. **Sampled V1, testing sensitivity to dataset size ("min500 max1200")**: As small datasets can be undersampled compared to large datasets, in this scenario, we sampled a minimum of 500 images per dataset and a maximum of 1,200 images per dataset. If the dataset was smaller than 500, we oversampled images randomly and added the duplicates to the dataset until we reached 500. We did not use all the data if the dataset was larger than 1,200. These numbers were chosen because when applying this sampling method to the 20 datasets, the total amount of labelled images was 15,257, similar to the total database (15,228 images).

3. **Sampled V2, testing sensitivity to dataset size ("750")**: In this sampling scenario, to sample equally from every dataset, 750 images from each were used. The total number of labelled images was 15,000, similar to the total database (15,228 images).

4. **Sampled V3, testing sensitivity to dataset size ("min500")**: To not delete data, in this scenario, only the small datasets were randomly oversampled to have a size of at least 500 images. As we kept all data in the other datasets, the total data size was larger than the baseline.

5. **Augmented V1, testing sensitivity to data augmentation ("mirrored rotated all")**: To increase the diversity of the data, the images were all used, and duplicates were mirrored and/or rotated. The rotation was kept between -15 and 15 degrees; in practice, the river almost always appeared at the frame's bottom. This was done because the data that must be analysed would normally also have the river on the bottom of the frame. In this scenario, the dataset contained twice the amount of images (30,456) as the baseline because a duplicate of each image was augmented randomly. Each image had a 50% chance to be mirrored. So approximately 7,614 images were mirrored, whilst all 15,228 copies were randomly rotated between -15 and 15 degrees.

6. **Augmented V2, testing sensitivity to data augmentation, ("mirrored rotated randomly")**: To increase the diversity of the data, the images were randomly selected to be mirrored and/or rotated. The rotation was kept between

-15 and 15 degrees. In 50% of the cases, an image was mirrored (a total of approximately 7,614), and in 50% of the cases, the image was mirrored randomly between -15 and 15 degrees. The dataset for this scenario consisted of 15,228 images.

7. **Augmented V3, testing sensitivity to data augmentation ("only mirrored")**: The images were ~~only randomly mirrored~~ randomly mirrored in 50% of the cases to disentangle the mirroring and rotation effect. The dataset for this scenario consisted of 15,228 images.

8. **Augmented V4, testing sensitivity to data augmentation ("only rotated")**: The images were ~~only~~ randomly rotated between -15 and 15 degrees ~~,~~ in 50% of the cases to disentangle the mirroring and rotation effect. The dataset for this scenario consisted of 15,228 images.

9. **Added V1, testing the sensitivity to data quality, added high definition non-floating wood**: In an attempt to increase the ~~model understanding of non-living~~ models' understanding of wood, we added photos of instream wood laying in the mostly dry riverbed ~~from an earlier survey~~ to the database. A total of 167 photos containing at least ~~1 example~~ one wood sample were added. The added data had pixel dimensions 4608 by 3456 and was ~~of~~ higher quality than the other 20 datasets (see Table 1). Here, the influence of ~~bbox~~ bounding box size and data quality ~~were tested~~ was evaluated.

10. **Added V2, testing the sensitivity to data quality and diversity, added 12 datasets**: ~~From~~ At a later stage in the testing process, from videos found online, a subset of the frames were labelled and added to the training database. A total of ~~12~~ 10 datasets ranging from 8 to ~~499~~ 118 images per dataset were added from locations in North America, New Zealand, and Switzerland. ~~The total amount of~~ Additionally, two more self-gathered datasets from different sources containing 207 and 499 images were added. As taken from the Internet, the 1206 images added in this scenario ~~was 1206, with~~ were compressed and have an average pixel resolution of ~~1650x1133. Also, as taken mainly from the internet, the data was compressed~~ 1650×1133. Therefore, the added data quality was worse than the original 20 datasets. The added data is indicated with the letter 'A' in figure 6. An 11 and A 12 are the self-gathered extra datasets.

11. **Removed, testing the sensitivity to data quality, removed worst performing datasets**: As lower quality data can weaken the models' understanding of wood, in this scenario, the quality of the data can be ~~analyzed~~ analysed by the effectiveness of the model trained in the base scenario to detect samples. ~~In this scenario, the two datasets at which the base model performed the worst were removed from the~~ The two datasets with the smallest relative bounding box size were assumed to include the fewest details and were of the lowest quality (see Figure 5). The three lowest-quality datasets were 12, 18, and 19. In this scenario, we removed datasets 12 and 19 from the training data to see whether the ~~detections on the other datasetsgot better.~~ other datasets' detections improved.

12. **Merged, testing the sensitivity to adding a time component, merged three images into 1**: Because often the distinction between a piece of instream wood or flow features like eddies and ~~wave ripples~~ waves was not clear from a single image, in this scenario we merged three images into one image after converting them to greyscale. Therefore, instead of

regular Red, Green, and Blue bands, the model was trained on greyscale images at T-1, T, and T+1, with T the timestep at which we were detecting. This was ~~hypothesized~~ hypothesised to aid the detection as waves and eddies change during a short timestep, whilst the wood ~~holds its shape~~does not.

13. **Double Resolution, testing the sensitivity to ~~increase~~ increasing the input image size to the model, from 416 to 832 pixels**: A CNN ~~is~~ was trained based on a specific pre-defined image resolution. As standard, images ~~are~~ were resized to a ~~416x416~~ 416×416 image resolution before they ~~are~~ were used in training and validation. Decreasing the images could result in ~~data loss~~losing details, especially in cases where the relative size of the wood pieces ~~were~~was small. Therefore, we ~~tested~~ evaluated the model's sensitivity to the input image size in this scenario by ~~performing the same tests with double the resolution (832x832)~~resizing the resolution to 832×832 pixels instead.

2.2.3 Model evaluation

Generally, a commonly used metric for object detection tasks to evaluate performance is mean Average Precision (mAP) (Tian et al., 2024), which combines three different measures: precision, recall, and intersection over union (IoU) (Zheng et al., 2020). Recall is the percentage of wood pieces detected by the algorithm out of all the logs that pass by. Precision indicates whether the piece the CNN detected ~~were~~was indeed instream wood. The object detection algorithm outputs either no bounding boxes or (multiple) bounding boxes for each image. Each bounding box indicates the outer limits of the object and has a confidence percentage corresponding to how certain the model is in its detection. More bounding boxes are classified as a detection when lowering the confidence threshold. Hence, the recall increases, and the precision decreases. The changes in precision and recall based on the threshold can be displayed in a precision-recall curve. The surface under the curve can be translated into a single average precision (AP) value for a specific IoU. However, this value does not compare different IoU thresholds. The IoU compares the label with detected bounding boxes by dividing their overlap by their combined total surface. For each IoU value, a different precision-recall curve can be created, resulting in different APs. When all different APs based on different thresholds and IoUs were combined into a single value, we got the mAP, which ranged between 0% and 100%. With an upper limit of 100% mAP, the model would have labelled every instance of instream large wood exactly as the humans that labelled the training data. However, as human labelling is imperfect, the mean Average Precision was still not an objectively perfect performance index.

Different applications of object detection call for different thresholds in recall, precision, and IoU, depending on the consequences. Depending on the large wood regime of a specific river, more emphasis can be ~~laid on either the recall or the~~ placed on either recall or precision. When the amount of wood passing is very low, e.g., one piece of wood a month, increasing recall can ensure that a piece is not missed. However, when using a ~~too high~~ too-high sensitivity, the model could wrongly detect wood in each frame, forcing the user to look at every image and delete all the false detections.

~~For this study, the same experiments were performed using 2000, 1000 and 500 augmented images for each of the 20 datasetsinitially. The performances were compared to test the model's sensitivity to the amount of images used. Then, for all of the 14 training scenarios we trained a modeland validated it in the six validation datasets. To mitigate the variance in~~

~~training results we performed the test three times and used the two best mAPs of every training instance to get robust results~~We trained a model for all 14 training scenarios and validated it with the six validation datasets. The training was performed in epochs, which represented the number of times all the training images were used to train the model. The model was validated using the validation data after a predetermined amount of epochs during the training process. The model's performance on the validation data was stored as mean Average Precision. During the same training session, the model can have several tests that perform similarly but also have an outlier in performance that cannot be reproduced. To account for this and not overestimate the performance, we used the best two validations mAPs to determine the performance of each training run. Furthermore, different training sessions using the same data can yield varying performance results, as the model may converge to different local optima depending on the initialization and training dynamics. Therefore, we run each test three times to compare different training scenarios. Hence, we obtained an average of six best mAPs for each scenario. After that, on a small subset of training scenarios, a newer YoloV7 model was trained to compare the results of different models on the same data. A final model was trained after determining which training strategies ~~work~~ worked best for which data types. This model was tested on ~~a test dataset~~ the test dataset described in section 2.1.4 that had never been used in any of the analysis and training efforts. In this way, the test dataset represented a case in which an unrelated wood monitoring study would use the model for out-of-the-box detection.

Finally, neural networks for object detection are often considered black boxes, which decreases their trustworthiness. To increase transparency, algorithms were developed to reverse the detection process and find the input pixels from the image weighted the highest when the process decided whether or not to detect an object. For the YoloV4 algorithm, we used a Python package called ~~"Yolime " was used~~ Yolime (Sejr et al., 2021) for this purpose. ~~Different~~ Yolime explains YOLOv4's object detection using LIME (Ribeiro et al., 2016). For each prediction, LIME perturbs the input data and observes prediction changes to highlight the image pixels that most influence the wood detection outcome. This process makes YOLOv4's predictions more understandable and, therefore, more trustworthy. Various instream wood samples from the database were hand-picked ~~, and an algorithm~~ and Yolime was run to determine which pixels ~~weighed the heaviest.~~

3 ~~Results and discussion~~

~~Our research led to the development of a comprehensive database containing labelled videos of instream~~ were most heavily used by the model to detect wood. The ~~labels cover a broad spectrum of sizes, particularly when comparing different datasets. They ranged from clearly identifiable downed trees with distinct features such as bark, branches, and brown colour to bounding boxes resembling small blobs that spanned only a few pixels in both dimensions. Also, distinguishing water waves and eddies from pieces of instream woodcan sometimes be challenging during labelling. Frequently, skipping through the frames provided a more precise understanding , as a wood piece would remain stationary in the frame while a wave would have dissipated. This observation suggested incorporating a temporal dimension could significantly enhance the detection process~~algorithm provided insights into which features of the image it identifies as characteristic of a piece of floating wood. Combined with data quality analysis, this understanding can help explain variations in model performance across different training scenarios.

3 Results

3.1 Training data: diverse but still clustered

~~Based on the analyses performed, the data utilized~~ The data used in this research appeared to be diverse. The PCA (Principal Component Analysis) yielded a low silhouette score ~~,~~ and a visual inspection of the t-SNE (t-Distributed Stochastic Neighbor Embedding) plot revealed only small clusters of similar data (see Supplementary Information), suggesting the presence of duplicates within the data. However, the analysis also uncovered similarities within each dataset. A comparison of the average bounding box sizes across datasets revealed distinct differences. Figure 5 illustrates the relative size of the bounding box compared to the overall image size, ~~highlighting significant~~ which was obtained by calculating the total amount of pixels in the envelope of the bounding box divided by the total amount of pixels in the image. The image highlights discrepancies in the sizes of labelled pieces of wood across different datasets. To adjust for the exponential distribution of calculated surface areas, we applied a square root transformation to the bounding box area for better visualization. The graph indicates that datasets 12, 18, and 19 ~~were either of lower quality (due to lower pixel resolution) or were positioned further from the stream. This suggested that the model may learn to recognize instream wood as large pieces within the camera frame, with distinct bark and other features easily associated with wood in great detail in some datasets , while in others, it might identify it as merely a few pixels clustered together, resembling a blob~~ of the original database had lower quality. This is the case because, in these cases, the size of the samples was small, and in the case of datasets 18 and 19, the camera was also located relatively far away. Examples of the difference in bounding box sizes between dataset 1 and dataset 12 are displayed in figures 13 and 14 in the supplementary material.

3.2 Training results: ~~fatabase vonfiguration~~ database configuration matters most

First, analyses showed that the model's performance did not increase by oversampling data from 500 (9,500) to 2000 (38,000) images per dataset. The best mAP was similar for both tests. Figure 6 shows the difference between a training instance with 2,000 augmented images per dataset on the left and 500 augmented images per dataset on the right. It shows that increasing the data for a training instance slows down the model convergence without increasing eventual mAP. ~~Testing the 14 different scenarios is hereafter done without the need for oversampling.~~

~~To test the model's sensitivity to stationary frames, dataset size, data augmentation and data quality, 13 different testing scenarios were performed.~~ Table 2 shows the training results of different scenarios. When using the total amount of 15,228 labelled images, the average performance in mAP was 63.42%~~, which was similar to the results that van Lieshout et al. (2020) found when creating a plastic detection algorithm. The table shows the average mAP of three training sessions started from the same pre-trained weights for the six representative datasets. The two best performing versions of the model were considered in each training session. Therefore, the displayed mAP in the first column is the average of six mAPs~~. The following columns show the difference in performance (mAP) of the base scenario compared to the scenarios based on the six best performances. At the bottom of the table, the average of the six ~~mAP~~ mAPs and the weighted average are shown. As there was a large variety

Figure 5. The relative size of the wood pieces compared to the image size per dataset. The relative size is represented by the square root of the surface of the bounding box sizes divided by the square root of the total image size. ~~To make the interpretation of the figure easier, we take the~~ The square root of the relative size is shown to facilitate the interpretation of the figure. The datasets from table 1 are indicated by the number. The 'A' indicates datasets that are added in scenario ~~12.~~ 10. The 'D' indicates the dataset added in scenario ~~11.~~ 9.
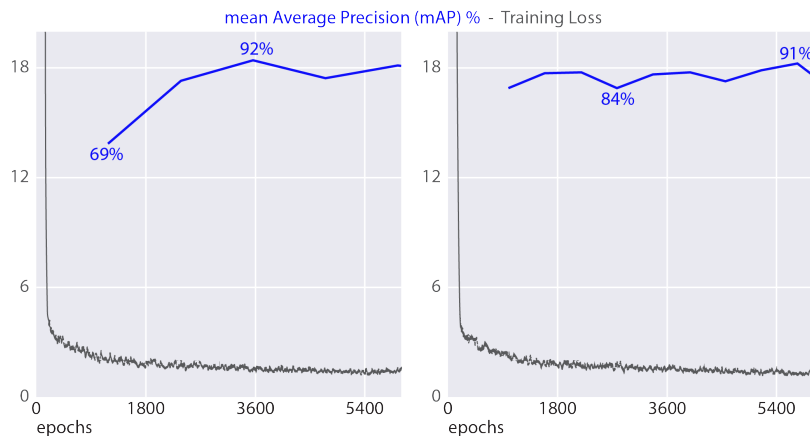
Figure 6. Training performance of dataset 13 (see Table 1) when using 2,000 images (left) and 500 images (right) per dataset (respectively 38,000 and 9,500 training images used). Blue: mean Average Precision, Grey: Complete Intersection-Over-Union Training Loss (Zheng et al., 2020)

in the sizes of the datasets, to not overestimate the importance of small datasets, the weighted average compensated for the

430   relative size of the datasets.

Dataset 20 ~~was small but had large variability incorporated in the images as they were taken from different sources. Therefore, the result from this dataset can be used to analyse~~ , despite its small size, exhibited large variability because the images were

Table 2. Neural Network performance for the base scenario in mean Average Precision (mAP, see section 2.2.3) ~~,~~ and relative change in mAP percentage points when using the 13 different training scenarios. mAP changes of more than ~~3 per cent~~ three percentage points, positive or negative, are indicated ~~in blue or red~~ with a star.

Scenario (dataset number)	Base Jpgs (mAP)	1 Trimmed	2 Sampled V1	3 Sampled V2	4 Sampled V3	5 Augmented V1	6 Augmented V2	7 Augmented V3	8 Augmented V4	9 Added V1	10 Added V2	11 Removed	12 Merged
Sorge Samsung Galaxy A4 (3)	80.33	0	-3.33-3.33*	-2.67	-1.33	-1.5	-2.17	-1.83	-2.5	-1.17	-1.83	-1.33	-18.83-18.83*
Sorge Pi 4 (7)	71.17	-0.17	-1.67	-3.67-3.67*	-0.83	-2.17	-3.83-3.83*	0.17	-1.67	-0.17	-1.17	-1	-3.17-3.17*
Borgne d'Arolla Pi 4 (11)	43.67	5-5*	14.67-14.67*	13.67-13.67*	15.33-15.33*	4.83-4.83*	0	0.5	-2.5	-1.17	-1	2.5	6.17-6.17*
Dixence Pi 4 (13)	92.83	0.5	-0.17	-1.67	0.67	1.17	2.33	-0.33	2.17	0.17	-0.67	0.17	1.17
Allier (18)	35.33	-0.5	-3.83-3.83*	-3.67-3.67*	-3.67-3.67*	-3.5-3.5*	-1.17	0.33	-1.83	-0.33	3-3	-19-19*	-1.5
Random Images (20)	57.17	-6.83-6.83*	2	3.33-3.33*	-4.67-4.67*	-11.33-11.33*	-9.83-9.83*	-1.17	-16.5-16.5*	22.83-22.83*	-13.83-13.83*	11.33-11.33*	**-**
Average	63.42	-0.33	1.28	0.89	0.92	-2.08	-2.45	-0.39	-3.81-3.81*	3.36-3.36*	-2.58	-1.22	**-**
Weighted Average	66.7	0.96	1.40	0.29	2.42	-0.50	-2.38	-.21	-1.94	-0.55	-1.14	-0.91	**-**

** dataset 20 does not contain a time series, and it is therefore not possible to merge

sourced from different locations. This variability makes this dataset particularly useful for assessing the model's ~~capacity to generalise the wood concept~~ ability to generalize the concept of wood and detect it across diverse conditions. On the other hand, the ~~weighted average was~~ larger datasets include cameras mounted to bridges and will, therefore, be a better representation of the primary use of the algorithm. Therefore, the weighted average is a better performance metric ~~when the wood detection we were interested in resembled the variety of the training data, which was, in this case, acquired mainly by cameras attached to a bridge~~ for the practical use of the algorithm.

Table 2 also shows the increase in model performance when changing the sampling strategy. When adding data from videos of floods containing instream wood found on YouTube and Twitter (scenario 10), the average mAP for all datasets went down by 2.58 ~~%~~ percentage points. This decrease in performance can arguably be attributed to the low quality of the data being confusing to the model. This observation was also strengthened by the ~~vast decrease in performance (-13.82%)~~ reduction in performance of this scenario when validating the high-definition random wood images dataset. ~~Also~~ Additionally, the model performed better when tested on the Allier River dataset, which ~~contains largely~~ mainly contains smaller (lower quality) samples of wood (see Figure 5). Instead of adding lower quality data, we added high-definition data of non-floating wood to the training data (Scenario 9), the ~~large~~ significant increase in performance when validating dataset 20 was explained by the algorithm generalizing the concept of wood. This increased the average performance but decreased the weighted average performance as the overall average label sizes of dataset 20 were relatively small.

~~With a significant difference between the best and the worst performing dataset~~

4   Discussion

4.1   Effect of data quality on performance and sampling strategies

Our research led to developing a comprehensive database containing labelled videos of instream wood. The labels covered a broad spectrum of sizes, particularly when comparing different datasets. They ranged from clearly identifiable downed trees with distinct features such as bark, branches, and brown colour to bounding boxes resembling less defined shapes that spanned only a few pixels in both dimensions. Additionally, distinguishing water waves and eddies from pieces of instream wood can sometimes be challenging during labelling, as an image is a snapshot. At that moment, a wave ripple can resemble a piece of wood.

The model's performance was slightly better than the results of approximately 50% mean precision that van Lieshout et al. (2020) found when creating a plastic detection algorithm and testing it on an unseen new location. The table shows the results of the different training scenarios explained in section 2.2.2. The sensitivity analysis indicated that increasing the dataset size to 2,000 images per set did not enhance performance compared to 500 images per set. This suggests that oversampling from a limited number of data sources does not improve model performance, as the model tends to overfit specific instances. Consequently, when training a custom model or making future enhancements, it is sufficient to utilize 500 labelled images. This threshold can optimise resource allocation and efficiency.

The proposed method does not require tuning of the algorithm for different sites, which allows its implementation on a range of different data sources. However, the training database, although extensive in size, is still limited in terms of the number of sources. The test where we see the best performance is during scenario 6 when using dataset 13. In this test, the model performed at a mean Average Precision (mAP) of 95.2%. The model's worst performance was on dataset 18 using scenario 11 at 16.3% mAP. From this large difference in performance, it can be argued that the ~~algorithm and data quality is the limiting factor~~ data quality and diversity are still limiting factors of the approach. This was reinforced by qualitative analysis of the ~~two worst-performing~~ quality of the datasets. As ~~this~~ lower-quality data might confuse the model, ~~another experiment~~ scenario 11 was performed where the datasets with lower ~~performance than 30 % mAP~~ quality (datasets 12 and 19) were excluded from the training data ~~(Scenario 11)~~. The results (table 2) ~~showed~~ show a weighted average decrease of 0.91 ~~%~~percentage points. This decrease was primarily linked to the worse performance of the model on the Allier River dataset (18). The decrease was large (19 percentage points) because the excluded dataset was taken from the same data source on a different day. Therefore, it can be argued that the model was still shown to overfit the training data even with the precautionary measures. Reasoning the other way around, ~~adding data (176 images in this case)~~ the -19 percentage points can be interpreted as +19 percentage points when adding data from the same ~~source but on a different day can increase model mAP by 19%~~scene on another day. This particular removed dataset contained 176 images. This shows a more practical implication for researchers. When starting a new monitoring project, ~~is it~~ it is good practice to label ~~a training dataset and add it~~ and add as few as 200 images to the larger database. In this way, one can train a site-specific wood detection algorithm which is shown to ~~be tens of percent~~ perform better than the model out of the box. The above findings also showed that although the validation data used to calculate the mAP was

taken from a different data source than the training data, there was still similarity. Data from the same camera on different days or the ~~exact~~ same location and date taken with different cameras were ~~not~~ shown not to be completely different.

485   Additionally, the results showed that the model's performance increased when using different sampling strategies (scenarios 2, 3 and 4). This was primarily because the detection on representative dataset 11 was more accurate. Dataset 10 was taken with the same camera angle at a different time. The scene looked different, with a sunny background and sharp shadows, compared to the evenly illuminating overcast of dataset 11. However, oversampling the 138 labelled images from dataset 10 positively affected the models' performance on dataset 11 in all three scenarios. This underlines the need for careful sourcing of training

490   datasets and shows the influence of dataset sizes. The model is trained to optimise performance on all training data and will, therefore, be biased towards the larger datasets. For the model to detect wood in a wide variety of scenes during the training phase, it is helpful to have equal dataset sizes, whilst a custom model for detecting wood from a single camera angle can benefit from oversampling from that specific scene. Furthermore, the data trimming from scenario 1 seemed to have limited effect.

The results of scenario 12, where we merged three frames into one to integrate a time component, yielded ~~exciting~~ interesting

495   insights. For datasets where the model already demonstrated robust performance, the accuracy experienced a noticeable decline. On the other hand, on datasets where the initial model struggled, ~~a remarkable improvement of nearly 10 per cent~~ an improvement of 6.2 percentage points was observed. This suggests that incorporating temporal information might be particularly beneficial when distinguishing between subtle features, such as ~~pices~~ pieces of wood and waves, proves challenging in a single frame. Further investigation into the ~~nuanced~~ impact of this temporal integration is needed to understand the specific

500   scenarios where this approach is advantageous. These findings underscore the potential of leveraging temporal information to improve river wood detection. ~~Lastly,~~

Another adjustment in the method was evaluated in scenario 13, where we doubled the image size after rescaling~~, shows increasing performance on 3 datasets specifically~~. The scenario demonstrated its greatest improvements in performance on the three datasets (7, 11, and 18) ~~. These are cases where the relative size of the wood examples is low~~ that have the lowest relative

505   bounding box sizes of the six representative datasets (see image 5)~~and can therefore be missedwhen decreasing the image size too significantly before running detections~~. This indicates that the reduction in image size was too extreme, and samples can be missed. For custom detection algorithms, it is advised to calculate the relative bounding box size of the samples on their specific location and optimize the image rescaling in terms of performance and computational efficiency.

After analysing the scenarios, a novel test dataset (see section 2.1.4) was introduced to evaluate the model's performance,

510   recorded during a flood event in the River Inn (see figure 7). Notably, the model had never used this dataset during its training phase, and no adjustments were made to hyperparameters based on this new data. The mean average precision (mAP) from the flood dataset was 61%, when using the base scenario. Increasing the input resolution of the model to 832×832, as per scenario 13, did not increase the performance (60.5% mAP). It is essential to highlight that this accuracy was achieved despite the flood event's challenging conditions and the imagery's relatively low-quality nature. Images with dimensions of

515   1280×720 pixels were captured using a mobile phone in timelapse mode. Furthermore, it was found that the model is better at detecting wood samples with larger bounding boxes. The ability of the model to identify larger wood elements is essential for its practical applicability. Large wood components often constitute a substantial proportion of total wood transport within rivers

Table 3. Comparison with YoloV7

Dataset No	mAP @ 0.5		
	YoloV4	YoloV7	Difference
1	80.33	77.18	-3.15
7	71.17	78.53	7.36
11	43.67	43.92	0.25
13	92.83	90.29	-2.54
18	35.33	21.07	-15.26
20	57.17	44.82	-12.35
Average	63.42	59.30	-4.12
Weighted Average	66.41	69.01	2.60

(Galia et al., 2018). Hence, combining our deep learning model's proficiency in detecting wood facilitates quantifying wood transport in river systems. The results suggest that the model can be used to estimate and monitor wood transport dynamics in
520 rivers, providing valuable insights into the ecological and geomorphic processes associated with fluvial environments. In cases with a particular interest in detecting the smaller samples, the limit in detectable size can be counteracted by increasing the image resolution or placing the camera closer to the stream.

4.2 Effect of neural network version on detection

The field of machine learning-based object detection moves fast. New versions of the existing state-of-the-art model are re-
525 leased every year. Therefore, we compared the performance of the 4th version of the You-Only-Look-Once model to the 7th (Wang et al., 2022) version. When comparing the training results on the same data using the base scenario, the results are shown in table 3. Even though the model became more efficient and smaller in terms of on-drive size (43% smaller from V4 to V7) and the ~~input image size~~ resolution to which the images were rescaled was larger (~~640x640~~ 640×640 for V7 and ~~416x416~~ 416×416 for V4), the performance did not drastically increase in our case. The average mAP went down by ~~4 per cent~~four
530 percentage points, whilst the weighted average went up by 2.5 ~~per cent~~percentage points, mainly because the model performed better on the largest dataset. ~~This indicates that~~ However, even though the newer model ~~can be trained~~ is demonstrated to perform better ~~for specific datasets. Still~~on conventional ML benchmarks (Wang et al., 2022) and in specific cases also has a higher mAP in our tests, when using a model without finetuning it to a particular study site, the 4th version of the YOLO model ~~performed better. According to Wang et al. (2022), with more conventional ML benchmarks, the performance of YoloV7 did~~
535 ~~increase. This can indicate that the training data's quality and/or diversity is still the training process's bottleneck in our setup.~~
   ~~These results show that even though the model is effective in detecting wood in rivers, the mean average precision is still dependent on the composition of the data used for training and is , therefore, still overfitting in some instances~~performs better. The differences in performance between the models are greater than those in many of the training scenarios. Therefore, the model choice is still essential in developing a wood detection algorithm.

540 ~~To evaluate the model's performance, a novel test dataset was introduced, captured during a flood event in the River Inn. Notably, this dataset had never been used by the model during its training phase, and no adjustments were made to hyperparameters based on this new data. The mean average precision (mAP) from the flood dataset was 61 per cent when using the base scenario. Increasing the input resolution of the model to 832x832, as per scenario 13, did not increase the performance (60.5% mAP). It is essential to highlight that this accuracy was achieved despite the flood event's challenging conditions and~~
545 ~~the imagery's relatively low-quality nature. Images with dimensions of 1280x720 pixels were captured using a mobile phone in timelapse mode. Furthermore, it was found that the model is better at detecting larger pieces of wood than smaller pieces. The ability of the model to identify larger wood elements is essential for its practical applicability. Large wood components often constitute a substantial proportion of total wood transport within rivers. Hence, combining our deep learning model's proficiency in detecting wood and georectification techniques facilitates the quantification of wood transport in river systems.~~
550 ~~The results suggest that the model can be used to estimate and monitor wood transport dynamics in rivers, providing valuable insights into the ecological and geomorphic processes associated with fluvial environments.~~
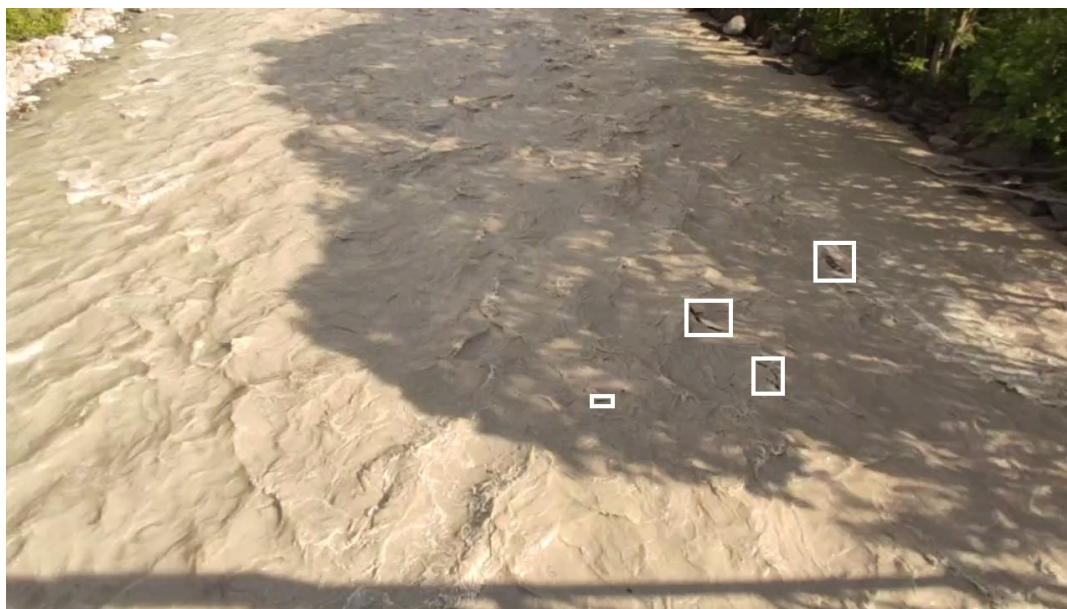
Figure 7. Example frame of the test dataset.

4.3   Understanding model predictions: wood features, surrounding water, and object size

The effectiveness of CNN has been displayed in various fields, such as security, transportation, and medical sciences (Kaur and Singh, 2023). However, ~~due to the model being~~ even though it is not, the model is often considered a black box, so its trustwor-
555 thiness can suffer. A CNN ~~is believed to make detections by understanding the concept of wood and use woodfeatures~~ takes statistical relationships in the pixel data and constructs characteristics to infer floating wood. In our case, these characteristics

are supposed to be characteristics of floating wood, like bark, ~~branches and colour to infer wood~~root wads, branch stumps, and surrounding water. However, ~~it might take shortcuts and~~ like a wolf-or-dog-classifier that was a snow detector (Ribeiro et al., 2016), it might use different characteristics of the training data to determine whether an object is wood. For instance, if a model is trained on data from a permanently mounted camera that constantly records the same scene, it can remember the scene and indicate everything out of the ordinary, like humans walking through the frame, to be large wood. If this is the case, the model would ~~struggle to perform~~ not demonstrate high performance on datasets that do not contain those characteristics. Therefore, it is essential to understand the model predictions.

We used ~~a picture~~ one of the pictures of instream wood from dataset 20 found online to ~~analyze~~ analyse which pixels in a figure were weighed the heaviest by the model to determine whether an object was a piece of instream wood. ~~Figure 8 shows~~ The pictures show a log stuck in a rapid, which has clear features of bark in brown colour with reflections and a fracture. Figure 8 demonstrates the inference of the image when using the base scenario as described in section 2.2.2 as compared to scenario 9. It also indicates the pixels the neural network uses to detect wood in the image. Remarkably, not just the pixels representing wood were indicated as applicable to detect instream wood. ~~The training data contained~~ In this case, the training data contains almost exclusively pieces of floating wood, and pieces on the bank that were not floating were not indicated. Therefore, the network seems to require the indication of water-containing waves next to the piece of wood to detect instream wood. ~~Also, in~~ In the base scenario, most ~~of the~~ training data contained small pieces of wood with a small relative bounding box size. Therefore, in the left image, the confidence of the model in the detected piece being wood is low, as the training data ~~did not contain a lot~~ lacked a sufficient number of similar high-definition images. In scenario 9 ~~,~~ however, high-definition images of non-floating wood were added to the training database ~~,~~ and therefore, the inference yields different results. This image resembles the added images; consequently, the piece was indicated as wood with a higher certainty. ~~Also, interestingly~~Interestingly, the model seems to ~~include~~ use pixels representing the wood (bark) texture and the fractured part for its detection. ~~These findings~~ This means that it detects bark and fracture features, and these findings would underscore the hypothesis in section 3.2 that there was a delicate balance between wood detection and small-object (~~blob~~less defined shapes) detection, primarily driven by the average size of samples in the training data.

4.4 Limitations of using low-cost cameras

Where low-cost cameras can aid research in offering economical means of capturing data, their use in detecting instream wood poses some limitations. Firstly, the lens and sensor quality of budget-friendly cameras often falls short compared to higher-end models, leading to less detailed images (Taylor et al., 2023). This lack of detail can make it even harder to distinguish small pieces of wood from noise within the frame (Casado-García et al., 2022). Additionally, the lighting conditions are generally handled less effectively, and glare from the water surface can obscure the visibility of wood. Lastly, in case of a lack of International Protection Rating certification, the lower durability of budget cameras in outdoor environments can lead to malfunctions and, eventually, gaps in the data.
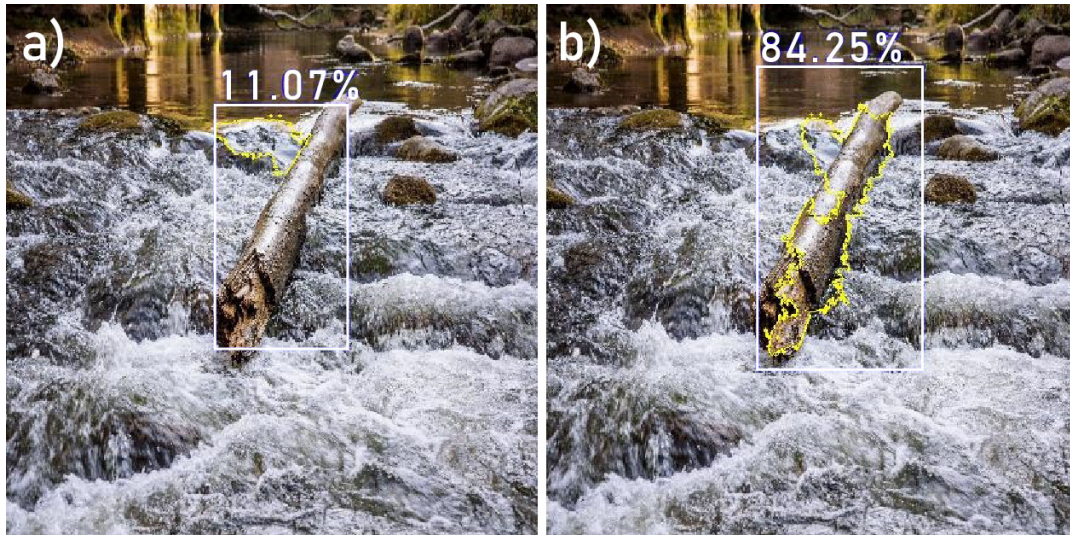
Figure 8. Wood detections according to ~~the models trained with the~~ a) Baseline scenario and b) Scenario 11: 'Added V1'. ~~In yellow, we~~ We indicate the heaviest weighting pixels in yellow according to the neural networks. Image source: dreamstime.com

5 Conclusions

590 We trained a Convolutional Neural Network to detect instream wood with a weighted average performance of 67~~per cent~~% mean Average Precision (mAP). On the best occasion, the model had a mAP of 93~~per cent~~% on one specific dataset. ~~It has to be noted that the~~ The performance was sensitive to the quality of the images in the training data, as concluded by a wide range of results. On an unseen test dataset, its ~~performance of~~ 61~~per cent mAP~~% mAP performance was in line with the results from the sensitivity tests. Efforts to improve the model's performance were, in some cases, successful. Depending on the data that

595 was used for training, ~~we enhanced up to a~~ the model's performance increased by up to 23~~per cent increase in mAP~~% (mAP). Changing the sampling strategy by adding or removing training data yielded considerable differences in average performance. ~~Also, although increasing in~~ Additionally, although enhancing the image input resolution ~~increases~~ increased the processing time and ~~makes~~ made the method more costly, in some instances, it did result in an almost 20 ~~percent~~ percentage points increase in mAP. On the other hand, data augmentation and different sampling methods did not seem to greatly influence the model's

600 performance.

Even though it was attempted to create a training database with various examples, the training results still ~~indicate~~ indicated the model overfitting the training data.

~~This study shows~~ Still, this study demonstrates that the model can generalise the concept of wood, ~~but primarily~~ mainly when training data consists of high-definition photos of ~~labeled wood . It still struggles to scale from high-definition images~~

605 ~~of wood to low-definition small samples . In the database, large examples~~ labelled wood samples. Additionally, more fitting to the general applicability of the method, we show it can also generalize the concept of wood in rivers when the samples have

different (smaller) dimensions. Large examples (of around 500x500 pixels) of wood were in the database, notably different from smaller samples (around 10x10 pixels). When training a custom model, it is advised to analyse the data that needs to be analysed and pick the datasets from our database accordingly. For this, it is crucial to use the training datasets that resemble it. A labelled training database of over 15,000 images was created in the research process. The training data is hosted publicly and can be used for future object detection refinements. Also, as the data is separated based on location and date, a customised model can be trained using the data that most closely represents the data of the person interested. For a new wood detection study, custom-labelled data can be added to the training database to increase the performance even more. This was underlined as tests demonstrated that adding only 176 labelled images of the same monitoring station but on a different day could increase the model's performance by 19 percentage points.

Despite its potential, the proposed method can not yet be used in real time. In future efforts, smaller versions of the evaluated models, like the Tiny version of the YOLO model, could be developed to run on in-field or mobile devices. Also, merging three subsequent frames in certain instances seemed to improve results. Hence, investigations into temporal imaging could improve the model as it benefits from including the time component of a video in the detection algorithm. Lastly, newly labelled datasets for custom models can be added to the larger database to aid in developing the performance of the model.

Code availability. https://github.com/janbertoo/Instream_Wood_Detection

Data availability. The data to which we have the rights is available at: 10.5281/zenodo.10822254 .

Author contributions. Study conception and design: JA, VRV
Data collection: JA, MV
Methodology design: JA, TB, VRV
Analysis and interpretation of results: JA, TB, VRV
Manuscript preparation: JA, VRV, TB.
All authors reviewed and approved the final version.

635 *Competing interests.* The authors declare that they have no conflict of interest.

Acknowledgements. This work has been supported by the Swiss National Science Foundation project PCEFP2186963 and the University of Lausanne. The data from the Allier River has been made possible by Véodis-3D consultancy because of their financial and technical assistance during the camera installation. We thank the comments from Prof. Iroumé, Prof. Diego Panici and Dr. Chris Tomsett, who helped us to improve the paper significantly.

References

Andreoli, A., Comiti, F., and Lenzi, M. A.: Characteristics, distribution and geomorphic role of large woody debris in a mountain stream of the Chilean Andes, Earth Surface Processes and Landforms, 32, 1675–1692, https://doi.org/10.1002/esp.1593, 2007.

Benda, L. E. and Sias, J. C.: A quantitative framework for evaluating the mass balance of in-stream organic debris, Forest Ecology and Management, 172, 1–16, https://doi.org/https://doi.org/10.1016/S0378-1127(01)00576-X, 2003.

Bengio, Y., Courville, A., and Vincent, P.: Representation learning: A review and new perspectives, IEEE Transactions on Pattern Analysis and Machine Intelligence, 35, 1798–1828, https://doi.org/10.1109/TPAMI.2013.50, 2013.

Bochkovskiy, A., Wang, C.-Y., and Liao, H.-Y. M.: YOLOv4: Optimal Speed and Accuracy of Object Detection, https://doi.org/10.48550/ARXIV.2004.10934, 2020.

Casado-García, A., Heras, J., Milella, A., and Marani, R.: Semi-supervised deep learning and low-cost cameras for the semantic segmentation of natural images in viticulture, Precision Agriculture, 23, 2001–2026, https://doi.org/https://doi.org/10.1007/s11119-022-09929-9, 2022.

Collins, B. D., Montgomery, D. R., Fetherston, K. L., and Abbe, T. B.: The floodplain large-wood cycle hypothesis: A mechanism for the physical and biotic structuring of temperate forested alluvial valleys in the North Pacific coastal ecoregion, Geomorphology, 139-140, 460–470, https://doi.org/10.1016/j.geomorph.2011.11.011, 2012.

Curran, J. H. and Wohl, E.: Large woody debris and flow resistance in step-pool channels, Cascade Range, Washington, Geomorphology, 51, 141–157, https://doi.org/10.1016/S0169-555X(02)00333-1, 2003.

Dalal, N. and Triggs, B.: Histograms of oriented gradients for human detection, in: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 1, pp. 886–893 vol. 1, https://doi.org/10.1109/CVPR.2005.177, 2005.

De Cicco, P. N., Paris, E., Ruiz-Villanueva, V., Solari, L., and Stoffel, M.: In-channel wood-related hazards at bridges: A review, River Research and Applications, 34, 617–628, https://doi.org/https://doi.org/10.1002/rra.3300, 2018.

Dibike, Y. and Solomatine, D.: River flow forecasting using artificial neural networks, Physics and Chemistry of the Earth, Part B: Hydrology, Oceans and Atmosphere, 26, 1–7, https://doi.org/https://doi.org/10.1016/S1464-1909(01)85005-X, 2001.

Diehl, T.: Potential Drift Accumulation at Bridges, 1997.

Duan, K., Bai, S., Xie, L., Qi, H., Huang, Q., and Tian, Q.: CenterNet: Keypoint triplets for object detection, Proceedings of the IEEE International Conference on Computer Vision, 2019-Octob, 6568–6577, https://doi.org/10.1109/ICCV.2019.00667, 2019.

Galia, T., Ruiz-Villanueva, V., Tichavský, R., Šilhán, K., Horáček, M., and Stoffel, M.: Characteristics and abundance of large and small instream wood in a Carpathian mixed-forest headwater basin, Forest Ecology and Management, 424, 468–482, https://doi.org/10.1016/j.foreco.2018.05.031, 2018.

Ghaffarian, H., Piégay, H., Lopez, D., Rivière, N., MacVicar, B., Antonio, A., and Mignot, E.: Video-monitoring of wood discharge: first inter-basin comparison and recommendations to install video cameras, Earth Surface Processes and Landforms, 45, 2219–2234, https://doi.org/10.1002/esp.4875, 2020.

Ghaffarian, H., Lemaire, P., Zhi, Z., Tougne, L., MacVicar, B., and Piégay, H.: Automated quantification of floating wood pieces in rivers from video monitoring: a new software tool and validation, Earth Surface Dynamics, 9, 519–537, https://doi.org/10.5194/esurf-2020-96, 2021.

Haschenburger, J. K. and Rice, S. P.: Changes in woody debris and bed material texture in a gravel-bed channel, Geomorphology, 60, 241–267, https://doi.org/10.1016/j.geomorph.2003.08.003, 2004.

Hassan, M. A., Hogan, D. L., Bird, S. A., May, C. L., Gomi, T., and Campbell, D.: Spatial and temporal dynamics of wood in headwater streams of the pacific northwest, Journal of the American Water Resources Association, 41, 899–919, https://doi.org/10.1111/j.1752-1688.2005.tb04469.x, 2005.

Hortobágyi, B., Vaudor, L., Ghaffarian, H., and Piégay, H.: Inter-basin comparison of wood flux using random forest modelling and repeated

680   wood extractions in unmonitored catchments, Hydrological Processes, 38, https://doi.org/10.1002/hyp.15176, 2024.

Innocenti, L., Bladé, E., Sanz-Ramos, M., Ruiz-Villanueva, V., Solari, L., and Aberle, J.: Two-Dimensional Numerical Modeling of Large Wood Transport in Bended Channels Considering Secondary Current Effects, Water Resources Research, 59, https://doi.org/https://doi.org/10.1029/2022WR034363, 2023.

Jodas, D. S., Brazolin, S., Yojo, T., de Lima, R. A., Velasco, G. D. N., Machado, A. R., and Papa, J. P.: A Deep Learning-based Ap-

685   proach for Tree Trunk Segmentation, in: 2021 34th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), pp. 370–377, https://doi.org/10.1109/SIBGRAPI54419.2021.00057, 2021.

Kalsotra, R. and Arora, S.: Background subtraction for moving object detection: explorations of recent developments and challenges, The Visual Computer, 38, https://doi.org/10.1007/s00371-021-02286-0, 2021.

Kaur, R. and Singh, S.: A comprehensive review of object detection with deep learning, Digital Signal Processing, 132, 2023.

690   Keller, E. A., MacDonald, A., Tally, T., and Merrit, N. J.: Effects of large organic debris on channel morphology and sediment storage in selected tributaries of Redwood Creek, northwestern California, US Geological Survey Professional Paper, 1454, 1–29, 1995.

Lassettre, N. S. and Kondolf, G. M.: Large woody debris in urban stream channels: Redefining the problem, River Research and Applications, 28, 1477–1487, https://doi.org/10.1002/rra.1538, 2012.

Lassettre, N. S., Piegay, H., Dufour, S., and Rollet, A.: Decadal changes in distribution and frequency of wood in a free meandering river, the

695   Ain River, France, Earth Surface Processes and Landforms, 33, 1098–1112, https://doi.org/10.1002/esp.1605, 2008.

Le Coz, J., Patalano, A., Collins, D., Guillén, N. F., García, C. M., Smart, G. M., Bind, J., Chiaverini, A., Le Boursicaud, R., Dramais, G., and Braud, I.: Crowdsourced data for flood hydrology: Feedback from recent citizen science projects in Argentina, France and New Zealand, Journal of Hydrology, 541, 766–777, https://doi.org/10.1016/j.jhydrol.2016.07.036, 2016.

Lecun, Y., Bengio, Y., and Hinton, G.: Deep learning, Nature, 521, 436–444, https://doi.org/10.1038/nature14539, 2015.

700   Lemaire, P., Piegay, H., MacVicar, B., Vaudor, L., Mouquet-Noppe, C., and Tougne, L.: An automatic video monitoring system for the visual quantification of driftwood in large rivers, III Wood in World Rivers, pp. 134–136, 2015.

Li, K., Wan, G., Cheng, G., Meng, L., and Han, J.: Object detection in optical remote sensing images: A survey and a new benchmark, ISPRS Journal of Photogrammetry and Remote Sensing, 159, 296–307, https://doi.org/https://doi.org/10.1016/j.isprsjprs.2019.11.023, 2020.

Lin, T. Y., Goyal, P., Girshick, R., He, K., and Dollar, P.: Focal Loss for Dense Object Detection, IEEE Transactions on Pattern Analysis and

705   Machine Intelligence, 42, 318–327, https://doi.org/10.1109/TPAMI.2018.2858826, 2020.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., and Berg, A. C.: SSD: Single shot multibox detector, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9905 LNCS, 21–37, https://doi.org/10.1007/978-3-319-46448-0_2, 2016.

Lucía, A., Comiti, F., Borga, M., Cavalli, M., and Marchi, L.: Dynamics of large wood during a flash flood in two mountain catchments,

710   Natural Hazards and Earth System Sciences, 15, 1741–1755, https://doi.org/10.5194/nhess-15-1741-2015, 2015.

Lyn, D., Cooper, T., and Yi, Y.-K.: Debris Accumulation at Bridge Crossings: Laboratory and Field Studies, Publication FHWA/IN/JTRP-2003/10. Joint Transportation Research Program, Indiana Department of Transportation and Purdue University, West Lafayette, Indiana, https://doi.org/https://doi.org/10.1016/j.ejrh.2023.101348, 2003.

MacVicar, B., Piegay, H., Henderson, A., Comiti, F., Oberlin, C., and Pecorari, E.: Quantifying the temporal dynamics of wood in large rivers: field trials of wood surveying, dating, tracking, and monitoring techniques, Earth Surface Processes and Landforms, 34, 2031–2046, https://doi.org/10.1002/esp.1888, 2009.

Pancini, D.: An Experimental and Numerical Approach to Modeling Large Wood Displacement in Rivers, Water Resources Research, 57, https://doi.org/https://doi.org/10.1029/2021WR029860, 2021.

Platts, W. S., Armour, C., Booth, G. B., Bryant, M., Bufford, J. L., Cuplin, P., Jensen, S., Lienkaemper, G. W., Wayne Minshall, G., Monsen, S. B., Nelson, R. L., Sedell, J. R., and Tuhy, J. S.: Methods for evaluating riparian habitats with applications to management., General Technical Report - US Department of Agriculture, Forest Service, 1987.

Prieto, A., Prieto, B., Ortigosa, E. M., Ros, E., Pelayo, F., Ortega, J., and Rojas, I.: Neural networks: An overview of early research, current frameworks and new challenges, Neurocomputing, 214, 242–268, 2016.

Pucci, A., Eickmeier, D., Sousa, H. S., Giresini, L., Matos, J. C., and Holst, R.: Fragility Analysis Based on Damaged Bridges during the 2021 Flood in Germany, Applied Sciences, 13, https://doi.org/10.3390/app131810454, 2023.

Ravazzolo, D., Mao, L., Picco, L., and Lenzi, M.: Tracking log displacement during floods in the Tagliamento River using RFID and GPS tracker devices, Geomorphology, 228, 226–233, https://doi.org/10.1016/j.geomorph.2014.09.012, 2015.

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A.: You Only Look Once: Unified, Real-Time Object Detection, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 779–788, https://doi.org/10.1109/CVPR.2016.91, 2016.

Ren, S., He, K., Girshick, R., and Sun, J.: Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, IEEE Transactions on Pattern Analysis and Machine Intelligence, 39, 1137–1149, https://doi.org/10.1109/TPAMI.2016.2577031, 2017.

Ribeiro, M. T., Singh, S., and Guestrin, C.: Why should I trust you? Explaining the predictions of any classifier, in: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1135–1144, ACM, 2016.

Ruiz-Villanueva, V., Wyżga, B., Mikuś, P., Hajdukiewicz, M., and Stoffel, M.: Large wood clogging during floods in a gravel-bed river: the Długopole bridge in the Czarny Dunajec River, Poland, Earth Surface Processes and Landforms, 42, 516–530, https://doi.org/https://doi.org/10.1002/esp.4091, 2017.

Ruiz-Villanueva, V., Mazzorana, B., Bladé, E., Bürkli, L., Iribarren-Anacona, P., Mao, L., Nakamura, F., Ravazzolo, D., Rickenmann, D., Sanz-Ramos, M., Stoffel, M., and Wohl, E.: Characterization of wood-laden flows in rivers, Earth Surface Processes and Landforms, 44, 1694–1709, https://doi.org/https://doi.org/10.1002/esp.4603, 2019.

Ruiz-Villanueva, V., Aarnink, J., Gibaja del Hoyo, J., Finch, B., and Vuaridel, M.: Integrating flow-, sediment- and instream wood-regimes during e-flows in the Spöl River (Swiss Alps), https://doi.org/10.3850/IAHR-39WC2521711920221000, 2022.

Sanhueza, D., Iroumé, A., Ulloa, H., Picco, L., and Ruiz-Villanueva, V.: Measurement and quantification of fluvial wood deposits using UAVs and structure from motion in the Blanco River (Chile), in: Proc. ofthe 5th IAHREurope Congress —NewChallenges inHydraulic ResearchandEngineering, edited by Aronne Armanini and Nucci, E., pp. 561–562, https://doi.org/10.3850/978-981-11-2731-1_216-cd, 2018.

Schenk, E. R., Mouline, B., Hupp, C. R., and Richter, J. M.: Large wood budget and transport dynamics on a large river using radio telemetry, Earth Surface Processes and Landforms, 39, https://doi.org/https://doi.org/10.1002/esp.3463, 2013.

Schwindt, S., Meisinger, L., Negreiros, B., Schneider, T., and Nowak, W.: Transfer learning achieves high recall for object classification in fluvial environments with limited data, Geomorphology, 455, 109 185, https://doi.org/https://doi.org/10.1016/j.geomorph.2024.109185, 2024.

Sejr, J. H., Schneider-Kamp, P., and Ayoub, N.: Surrogate Object Detection Explainer (SODEx) with YOLOv4 and LIME, Machine Learning and Knowledge Extraction, 3, 662–671, https://doi.org/10.3390/make3030033, 2021.

Sokolova, M., Cordova, M., Nap, H., van Helmond, A., Mans, M., Vroegop, A., Mencarelli, A., and Kootstra, G.: An integrated end-to-end deep neural network for automated detection of discarded fish species and their weight estimation, ICES Journal of Marine Science, 80, 1911–1922, https://doi.org/10.1093/icesjms/fsad118, 2023.

Sun, R., Lei, T., Chen, Q., Wang, Z., Du, X., Zhao, W., and Nandi, A. K.: Survey of Image Edge Detection, Frontiers in Signal Processing, 2, https://doi.org/10.3389/frsip.2022.826967, 2022.

Swaroop, P. and Sharma, N.: An Overview of Various Template Matching Methodologies in Image Processing, International Journal of Computer Applications, 153, 8–14, https://doi.org/10.5120/ijca2016912165, 2016.

Taskesen, E.: Python package clustimage is for unsupervised clustering of images., https://erdogant.github.io/clustimage, 2021.

Taylor, L. S., Quincey, D. J., and Smith, M. W.: Evaluation of low-cost Raspberry Pi sensors for structure-from-motion reconstructions of glacier calving fronts, Natural Hazards and Earth System Sciences, 23, 329–341, https://doi.org/10.5194/nhess-23-329-2023, 2023.

Tian, J., Jin, Q., Wang, Y., Yang, J., Zhang, S., and Sun, D.: Performance analysis of deep learning-based object detection algorithms on COCO benchmark: a comparative study, Journal of Engineering and Applied Science, https://api.semanticscholar.org/CorpusID:268631068, 2024.

Van Der Maaten, L. and Hinton, G.: Visualizing data using t-SNE, Journal of Machine Learning Research, 9, 2579–2625, 2008.

van Lieshout, C., van Oeveren, K., van Emmerik, T., and Postma, E.: Automated River Plastic Monitoring Using Deep Learning and Cameras, Earth and Space Science, 7, https://doi.org/10.1029/2019EA000960, 2020.

Viso.ai: Viso Suite: The One No Code Computer Vision Platform, 2022.

Wang, C.-Y., Bochkovskiy, A., and Liao, H.-Y. M.: YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, 2022.

Wohl, E.: A legacy of absence: Wood removal in US rivers, Progress in Physical Geography, 38, 637–663, https://doi.org/10.1177/0309133314548091, 2014.

Wohl, E., Lininger, K. B., Fox, M., Baillie, B. R., and Erskine, W. D.: Instream large wood loads across bioclimatic regions, Forest Ecology and Management, 404, 370–380, https://doi.org/10.1016/j.foreco.2017.09.013, 2017.

Wohl, E., Scott, D. N., and Lininger, K. B.: Spatial Distribution of Channel and Floodplain Large Wood in Forested River Corridors of the Northern Rockies, Water Resources Research, 54, 7879–7892, https://doi.org/10.1029/2018WR022750, 2018.

Wohl, E., Kramer, N., Ruiz-Villanueva, V., Scott, D. N., Comiti, F., Gurnell, A. M., Piegay, H., Lininger, K. B., Jaeger, K. L., Walters, D. M., and Fausch, K. D.: The natural wood regime in rivers, BioScience, 69, 259–273, https://doi.org/10.1093/biosci/biz013, 2019.

Xu, Y. and Goodacre, R.: On Splitting Training and Validation Set: A Comparative Study of Cross-Validation, Bootstrap and Systematic Sampling for Estimating the Generalization Performance of Supervised Learning, Journal of Analysis and Testing, 2, 249–262, 2018.

Zhang, Z., Ghaffarian, H., Macvicar, B., Vaudor, L., Antonio, A., Michel, K., and Piégay, H.: Video monitoring of in-channel wood: From flux characterization and prediction to recommendations to equip stations, Earth Surface Processes and Landforms, 46, 822–836, https://doi.org/10.1002/esp.5068, 2021.

Zhao, Z. Q., Zheng, P., Xu, S. T., and Wu, X.: Object Detection with Deep Learning: A Review, IEEE Transactions on Neural Networks and Learning Systems, 30, 3212–3232, https://doi.org/10.1109/TNNLS.2018.2876865, 2019.

Zheng, Z., Wang, P., Liu, W., Li, J., Ye, R., and Ren, D.: Distance-IoU Loss: Faster and Better Learning for Bounding Box Regression, AAAI, 34, 2020.

Zoph, B., Cubuk, E. D., Ghiasi, G., Lin, T.-Y., Shlens, J., and Le, Q. V.: Learning Data Augmentation Strategies for Object Detection, in: Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII, p. 566–583, Springer-Verlag, Berlin, Heidelberg, https://doi.org/10.1007/978-3-030-58583-9_34, 2020.

Zou, X.: A Review of Object Detection Techniques, in: 2019 International Conference on Smart Grid and Electrical Automation (ICSGEA), pp. 251–254, https://doi.org/10.1109/ICSGEA.2019.00065, 2019.

Àlex Solé Gómez, Scandolo, L., and Eisemann, E.: A learning approach for river debris detection, International Journal of Applied Earth Observation and Geoinformation, 107, 102 682, https://doi.org/https://doi.org/10.1016/j.jag.2022.102682, 2022.

 Data diversity

Figure 12 presents a visualization of the data diversity using T-distributed Stochastic Neighbor Embedding (t-SNE), a dimensionality reduction technique primarily employed for visualization purposes. The 20 known datasets are represented as distinct clusters, with each example marked by its cluster's colour. While the absolute distances between examples in the plot are not meaningful, the method does cluster similar neighbours closer together. The visualization demonstrates that, in general, the samples are
815   well distributed. The overlap between clusters accounts for the low silhouette score, indicating high variability within the data. However, small, concentrated groups of images outside the central cluster can be identified as duplicates in the training data. To address this, the data-trimming step will aim to reduce the influence of these sub-clusters. This will prevent the final model from being disproportionately rewarded for correctly detecting a specific piece of wood, thus mitigating the risk of overfitting.
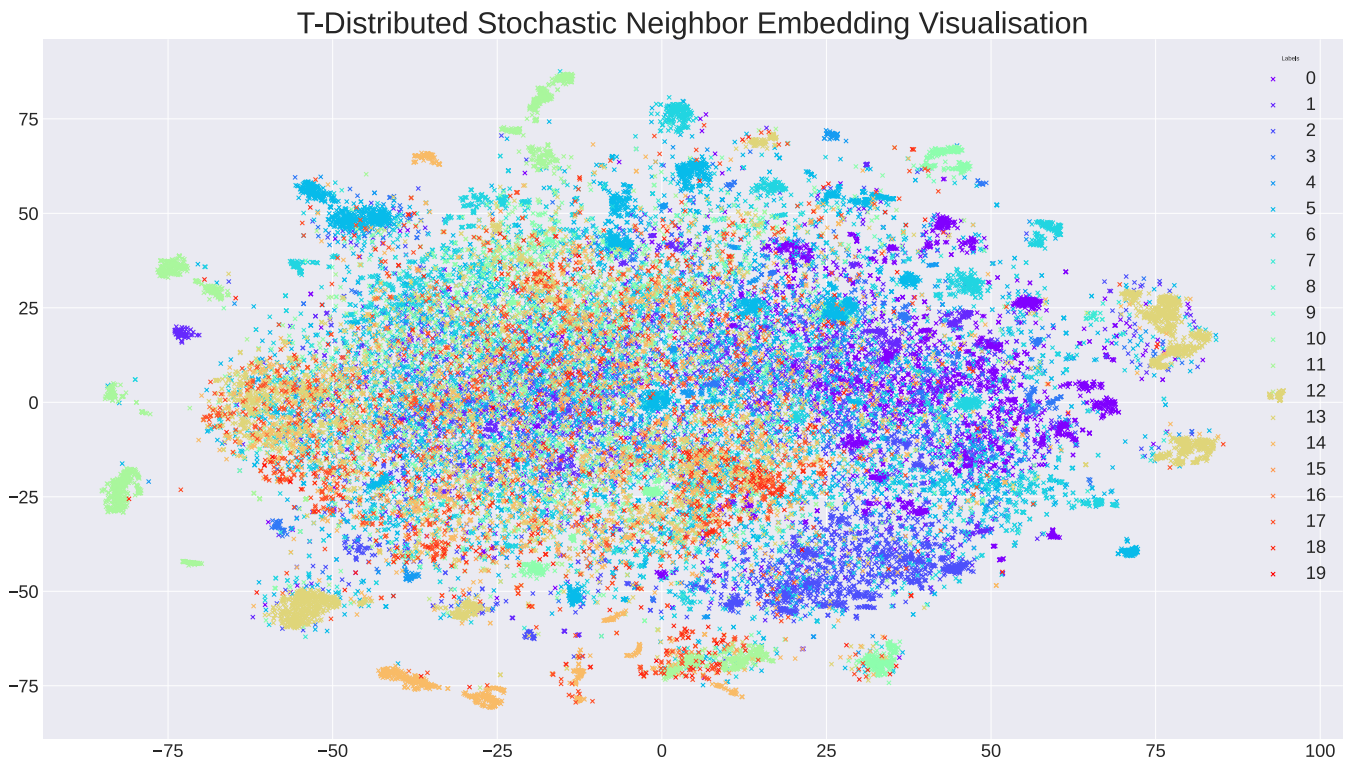
Figure 12. Clustering visualized using T-Distributed Stochastic Neighbor Embedding (Van Der Maaten and Hinton, 2008). Visualisation of all 33,000 samples. Different colours represent the 20 different datasets. Distinctive clusters in the figure have mainly the same colour and are, therefore, part of the same dataset.

Dataset examples

820   Figure 13 and 14 show examples of bounding boxes in datasets 1 and 12. The samples in dataset 12 are small and are therefore cropped and increased in size by 500%.

Figure 13. Example image dataset 1. The bounding boxes are cropped without resizing.
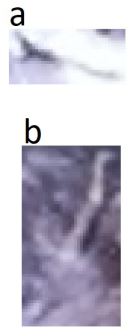
Figure 14. Example image dataset 12. The bounding boxes are cropped and uniformly resized by 500%.