Enhancing Water Level Identification with a Barcode-Patterned Panel and Machine Learning*

Gabriel Montagni Domingues Filho[1], Caetano Mazzoni Ranieri[1], Douglas Queiroz Galucio Batista[1], and Jó Ueyama[1]

Institute of Mathematical and Computer Sciences, University of São Paulo, Av. Trab. São Carlense, 400 - Parque Arnold Schimidt, 13566-590, São Carlos, SP Brazil
{domingues.gabriel; cmranieri; douglas.queiroz}@usp.br, joueyama@icmc.usp.br

Abstract. Floods cause significant material and human losses worldwide, leading to research in monitoring water levels in urban streams. Existing technologies, such as pressure and ultrasonic sensors, are accurate but costly to deploy. Although ground cameras offer a low-cost alternative, current approaches relying on weak visual markers are sensitive to environmental factors. We address this research gap by introducing a physical marker, called "barcode panel", which is a stainless steel plate with printed black stripes indicating water level. A deep learning algorithm was employed to accurately predict water levels based on this marker. We evaluated our approach using two datasets: one in a pool and another in an actual river. The model demonstrated precise water level predictions in the pool dataset and good results for the river dataset, despite training solely on the pool images. These promising results provide valuable insights for further studies and practical applications.

Keywords: computer vision · deep learning · flood management · visual marker · water gauge

1 Introduction

Floods pose a critical concern in Brazil, affecting a significant number of municipalities, with over 825 of them being highly vulnerable to landslides and flash floods. Particularly in the densely populated southeast region, these disasters account for a staggering 90% of all registered natural disasters. The financial toll of these events on Brazil between January 1, 2013, and April 5, 2022, is estimated to be a staggering USD 67 billion [17,16]. According to the Brazilian National Water Agency (ANA), the impact on communities is substantial, as evidenced by the fact that more than 800,000 individuals were affected by floods

* Supported by the São Paulo Research Foundation (FAPESP), grants 2023/00017-2, 2021/10921-2, 2013/07375-0, and 2022/09644-7, the Coordination for the Improvement of Higher Education Personnel (CAPES) and the National Council for Scientific and Technological Development (CNPq).

in 2020. The situation worsened in 2021, with the number of affected people exceeding 1 million [4].

One crucial step in addressing this problem is identifying water levels in natural water bodies like creeks and rivers, which enables the estimation of flood risks and the implementation of damage control measures. Various data sources can be utilized to determine water levels in water streams. The first approach involves using submersible pressure sensors [7]. While these sensors are relatively easier to deploy, floods often carry sediments and waste that can damage them [28]. Alternatively, sensors that operate without direct contact with the water can be employed. Ultrasonic sensors provide one such possibility [18]. Additionally, meteorological measurements [5] or cameras that capture river images can be used to infer water levels [9].

Numerous studies utilize pattern recognition and computer vision techniques [29]. These studies can be broadly categorized into flood modeling, flood inundation mapping, flood monitoring, and early warning systems [12]. In flood modeling, research efforts are focused on understanding hydrological flow behaviors [24,28] and designing hydrological structures to mitigate floods [21]. Flood inundation mapping entails the determination of water surfaces or the estimation of water depth. Some approaches leverage spatial images [2,3,22], while others employ deep learning techniques for flood hazard mapping and water segmentation [14,25]. Flood monitoring and early warning systems encompass forecasting methodologies [8,10] and water level measurement techniques [15,29].

When it comes to utilizing images captured by ground cameras positioned near water bodies, machine learning-based models tend to surpass traditional image processing techniques due to the nonlinearity inherent in flood events [13]. Recent studies in this domain have embraced this paradigm and explored various approaches, including deep learning [26,27], multiclass segmentation [20], and classical machine learning models [1]. These methods have emerged as prominent solutions for flood detection and analysis, and their advancement continues to accelerate, with widespread adoption in the field [6].

However, the existing literature has not extensively explored using visual markers as references to improve the accuracy of water measurement systems. Instead, most studies have primarily focused on estimating water levels directly from images. In cases where panels or water gauges were used, image processing techniques were employed to infer the water level. This approach proved to be sensitive to factors such as variations in camera pose, weather conditions, and potential data drift.

We present a novel method for estimating water levels, which utilizes a physical, visual marker placed within the water, a ground camera, and a deep learning algorithm. The visual marker, known as "barcode panel", is a stainless steel panel featuring a pattern of evenly spaced, horizontal black rectangles arranged like a barcode. The deep learning model was designed to accurately count the number of black rectangles visible above the water's surface, allowing a damage control system to identify the water level.

We built two datasets to evaluate such an approach: one collected in a pool, and the other, in an actual river. The pool dataset was used to train a model based on the ResNetV2 neural network [11]. We evaluated this dataset not only on the pool dataset but also using the river dataset.

We framed the task as either a regression or a classification problem and compared the performance of both approaches. We also evaluated the effect of cropping the barcode from the image before feeding it to the neural network, which could be achieved using a separate object detection model arranged in a cascade approach (i.e., the output of the object detection model serves as input to the regression or classification model that counts the bars). Results have shown that this approach can generalize different environments and camera positionings.

The remainder of this article is organized as follows. Section 2 provides an overview of related papers, methodologies, and applications. Section 3 details the barcode approach, datasets, networks, and training-validation method used in our study. The results of the experiments are presented and discussed in Section 4 and discussed in Section 5. Conclusions are drawn in Section 6, along with directions for future work.

2 Related Work

Many works and approaches have been developed to predict floods and mitigate the losses incurred due to these devastating events. In this section, we examine the literature around machine learning models and computer vision approaches.

Furquim *et al.* [10] proposed the development of a remote sensor network installed in an urban river. The proposed system captured water level images using ground cameras and inferred its water depth using pressure sensors. The obtained data served to study the use of several machine learning techniques, such as Random Forest, Random Tree, Best-First Decision Tree, Simple Cart, Multi-Layer Perceptron, and Bayesian Learning. Although accurate results were yielded, the authors observed that maintaining all the equipment and classification algorithms functioning is costly in terms of resources and computational power. Additionally, pressure sensors must be periodically recalibrated due to water action, which incurs maintenance costs.

Aljohani *et al.* [1] proposed a framework with multiple machine learning models, such as Random Forest, Decision Tree, K-Nearest Neighbor, Support Vector Machine, Logistic Regression, and Deep Learning. Among these, Random Forest and Decision Tree had the best accuracy, and they aim to use the framework to provide early flood detection. In Park *et al.* [20], a flood detection model based on transformers was proposed using the SpaceNet 8 dataset. The results were superior to models based on convolutional neural networks (CNNs). WU *et al.* [27] utilized SAR images in a deep learning-based flood detection model for segmentation. They concluded that CNNs have great potential for flood detection in near-real-time flood prediction.

Lin *et al.* [15] used surveillance cameras present in rivers and hydraulic installations to obtain images of the rivers. From this, the authors applied image processing techniques to identify the water level in the meter through co-linearity equations. The method demonstrated an error rate of 0.01 m using images from only one camera and could handle induced camera motion and noise caused by rain, which brought reliable results. However, if changes in the camera are unexpected and the weather conditions are too extreme, the detection is compromised.

Zhang *et al.* [29] proposed a water level monitoring system using existing surveillance cameras, which reduces the cost of the application. The authors assumed that the water level is usually located where the local gray color change is the largest on the water meter. The image processing system is based on the maximum mean difference of gray and edges. Real-world experiments showed an accuracy of 90%. However, the experiments revealed issues arising from the concentration of floating debris around the water meter during river floods, making it impossible to visualize the markers. The system also faced issues caused by water surface vibrations, resulting in blurred capture of the target and the water line.

Vandaele *et al.* [25] developed a deep learning-based approach using automated semantic water segmentation to facilitate the process of estimating river levels from camera images. The dataset used in the study was obtained from the Severn and Avon rivers in the United Kingdom. The results were compared with measurements taken by nearby water gauges. The study concluded that the approach could provide an easy and cost-effective way of monitoring river levels, especially in locations where water gauges or other detection mechanisms are unavailable.

Pan *et al.* [19] developed a low-cost surveillance system with measurement stations and a monitoring center. They used video cameras, water level analyzers, and wireless connections. They also used three methods to evaluate the system: the difference method, dictionary learning, and deep learning. Finally, they show that the deep learning method based on convolutional neural networks (CNN) presents the best result regarding accuracy and stability.

Sabbatini *et al.* [23] proposed a water level monitoring solution based on image processing of a staff gauge using mechanisms of automatic computer vision. To achieve this, the system was divided into two parts, the first of which classified the image as night, day, or poor quality. The second part extracted information about the water level. The authors obtained satisfactory results with nighttime images, but they encountered difficulties dealing with the varying sunlight angles during the day.

The studies cited in the literature showcase diverse approaches for detecting water levels and recognizing or predicting floods. While a few solutions have utilized water gauges as references, these approaches are location-specific and lack generalizability across different environments. In our work, we propose a novel approach that combines a new reference marker called barcode panel with deep learning techniques that were not used in this kind of task. By integrating these elements, our method offers potential advantages in addressing challenges related to weather variations and sunlight incidence that commonly affect existing approaches. With this innovative solution, we aim to advance the field of water level detection and flood prediction, contributing to developing a more adaptable and robust system using an approach that was not totally explored and techniques other than the cited ones.

3 Our barcode panel approach to detecting floods

As already stated, this study aimed to assess the effectiveness of ResNetV2-based models on two distinct datasets for computer vision-based water level measurement. The datasets comprise of water bodies featuring a barcode panel (refer to Section 3.1), in which the number of bars visible above the water surface is directly proportional to the water level.

The study examined two problem formulations: (i) a classification task, treating the possible numbers of bars as individual categories, and (ii) a regression task, estimating the number of bars visible above the water surface as a single real number. The following subsections provide detailed information about the barcode panel, the dataset, and the method employed.

3.1 The Barcode Panel

A schematic representation of the barcode panel deployed to a water body is shown in Figure 1. The barcode panel comprises a stainless steel surface with a "zebra" pattern printed on it, consisting of black bars along the plate. When the barcode panel is placed in a river, some of the bars remain above the surface and are used as a reference to measure the water level. This approach is expected to be applicable in various bodies of water and regions, requiring only a camera to capture images and reducing the need for prior knowledge about the specific region. The barcode panel used in the experiments consists of 7 black bars with a spacing of 15 cm between each one, and each bar is 5 cm in height and 20 cm in width. Therefore, the barcode panel has dimensions of 155 cm x 40 cm.

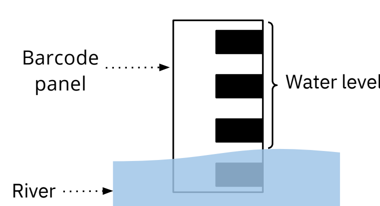


Fig. 1: Barcode panel application. Three of the four bars are above the water surface, indicating the water level.

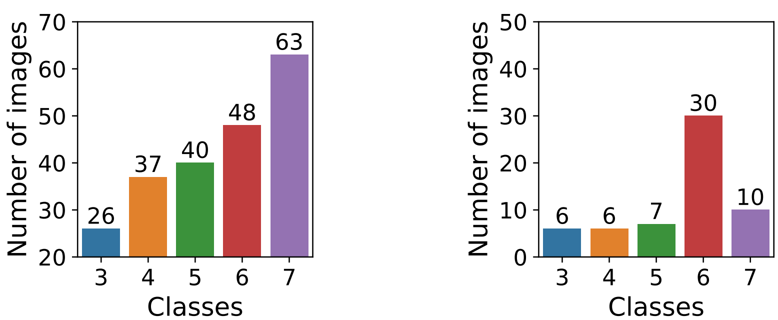
3.2 Constructed datasets

To initiate our work, we gathered two distinct datasets[1]. In the first dataset, we immersed the barcode panel in a pool, intentionally altering the water levels by adjusting the panel's position through upward or downward movements. We captured images from diverse perspectives, incorporating different positions and angles. In order to introduce varying environmental conditions, we also simulated artificial rain by spraying water using a hose during specific image captures. This deliberate approach aimed to test the models' ability to accurately identify the barcode under different environmental influences. Besides, in both datasets, the images were captured during the daytime.

We could simulate different water levels by varying the position of the barcode, which could be done by manipulating a supporting platform placed below it. We gathered images within a range of 3 to 7 visible bars, as constrained by the pool depth. We annotated the images by providing a label (i.e., the number of visible bars) and a bounding box surrounding the panel.

The second dataset was created with images acquired in an actual river, the Maués-Açu River at Maués-AM, Brazil. Similar to the first dataset, the images were captured from different positions, angles, and water levels.

The resulting pool dataset contains 214 images, and the river dataset contains 59 images. Fig. 2 shows the distribution of classes in the datasets, and Fig. 3 provides a sample image of which one.



a) Pool dataset

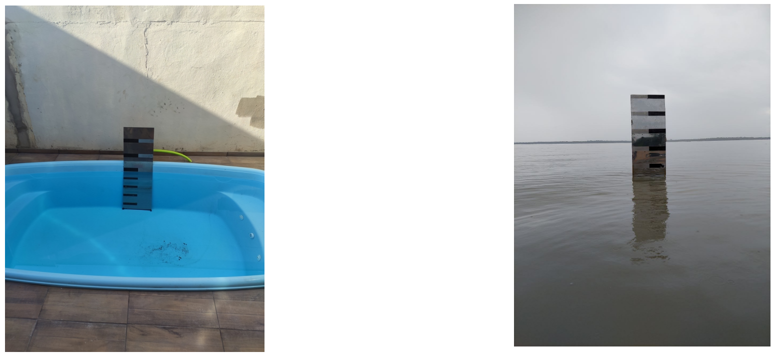
b) River dataset

Fig. 2: Classes distribution over the datasets.

3.3 Implementation of AI models

We approached the problem from two perspectives. Firstly, we treated it as a regression problem, aiming to predict the continuous value of the number of bars

[1] `https://github.com/domingues100/Water-Level-Detection-Datasets`



a) Sample of the pool dataset. b) Sample of the river dataset.

Fig. 3: Sample images from both datasets. It's worth noting that the background environments in both datasets exhibit notable differences, as does the water turbidity.

above the water. Subsequently, we reframed the problem as a classification task, where the labels corresponded to integers from 3 to 7, representing the number of bars above the water.

The model underwent training using the pool dataset, which offered a more controlled environment and a larger volume of images. Additionally, in anticipation of potential applications of the proposed models in conjunction with object detection models, we also trained a separate model utilizing cropped images specifically focused on the panel.

Therefore, we analyzed two factors with two conditions each: (i) the problem formulation (i.e., classification or regression), and (ii) the extent of the input image (i.e., complete or cropped). This experiment resulted in four trained models. As the river dataset was used to validate whether the models trained on the pool dataset could generalize to a different environment, we got eight possible cases to be analyzed.

We employed a ResNetV2 [11] with pre-trained weights from the ImageNet dataset [22], a widely used dataset for pre-training deep learning models. The pre-trained weights were used to initialize the model, which was then fine-tuned on our datasets. To use this model in the regression approach, the final layer was removed and changed by a single unit with linear activation. For the classification approach, we treated the labels, corresponding to the number of visible bars, as independent categories. In computational terms, the network comprises 56 million parameters, 17 billion FLOPs (floating-point operations), and a size of 213.41 MB. If the images are cropped and resized, the required computational power can be increased. Fig. 4 illustrates the proposal.

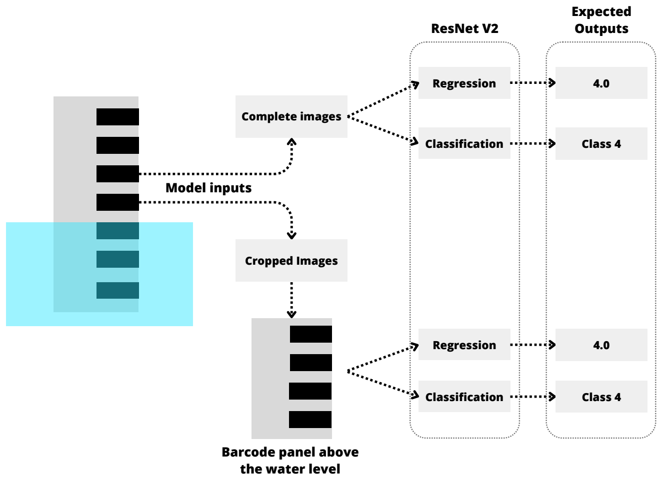


Fig. 4: Diagram illustrating the proposed approach. Our methodology involved utilizing two types of inputs: complete images and cropped images focused on the barcode panel. We formulated the task as both a classification and a regression model for each input type, resulting in four distinct scenarios.

3.4 Training and Validation

We adopted a k-Fold Stratified method commonly used to train the models. This approach ensures that each fold has a proportional representation of the different classes, which is important to avoid overfitting and bias. In k-Fold Stratified, the dataset is divided into K folds, and each fold is used once as a validation set while the $k-1$ remaining folds are used for training. This process is repeated k times, with each fold being used as a validation set once.

Fig. 5 presents a schematic representation of the procedure. By doing so, we can maximize the use of available data for both training and validation. Furthermore, this approach provides a more reliable estimate of the model's performance than a standard hold-out approach, since we obtain k estimates of each performance metric. The final performance metric is the average of these k estimates.

3.5 Experimental Setup

The experimental setup was carefully designed to enable direct comparisons between the eight cases of our study and other similar research studies reported in the literature.

The Keras framework with Tensorflow backend was used for all experiments. The training process was performed on Google Colaboratory, providing a convenient environment and a free GPU.

The models were trained for 100 epochs with batches of size 32. The images were resized to 224x224 pixels, and to optimize the model, we used the Adam

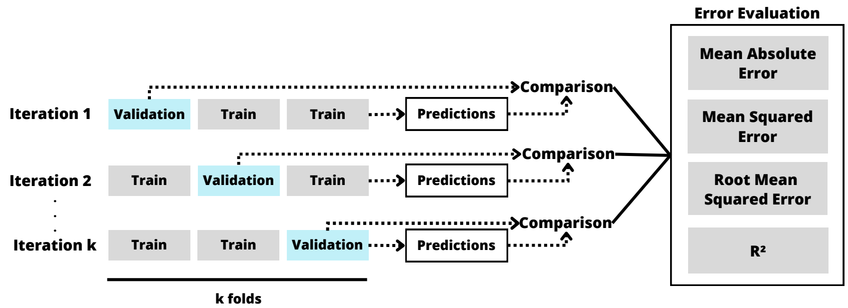


Fig. 5: Illustration of the Stratified K-Fold paradigm. The ground-truth labels of the validation folds are compared with the predicted outputs, allowing for error evaluation.

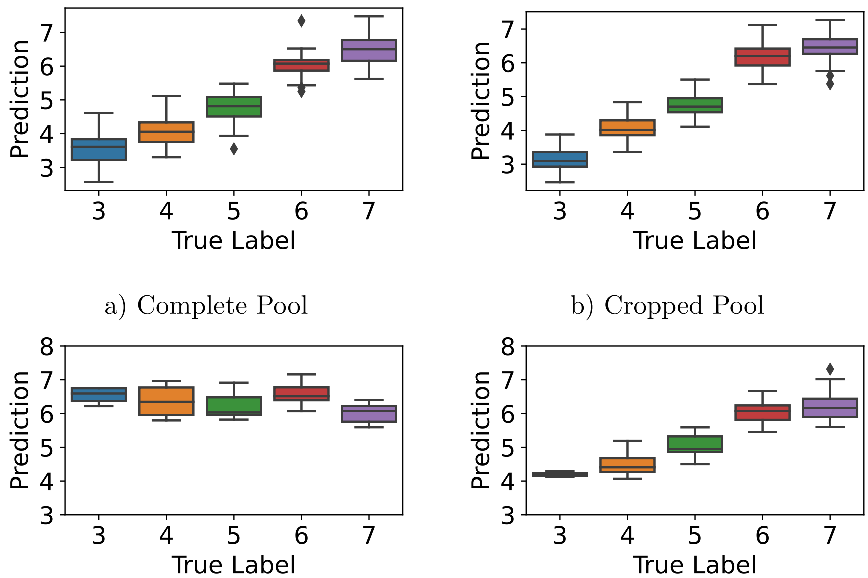
optimization algorithm, with a learning rate of 10^{-3}. The loss function used was mean squared error (MSE). Furthermore, for the stratified K-Fold, the number of splits was set as 10, and the data was shuffled. Emphasizing that the same folds were used for all tests performed, allowing for direct comparisons between the performance of each.

4 Results

To analyze our classification models and enable comparison with other results in the literature, we computed the metrics shown in Table 1. Furthermore, to visualize the distribution of the predictions and identify outliers, we generated boxplots to illustrate the data distribution for each ground-truth value for both regression models, as shown in Figure 6.

Table 1: Performance metrics for classification models with and without cropped images, compared across both datasets.

Pool Dataset												
	Model without crop						**Model with crop**					
	3	**4**	**5**	**6**	**7**	**Mean**	**3**	**4**	**5**	**6**	**7**	**Mean**
Precision	0.93	0.86	0.81	0.89	0.83	0.864	1.00	0.93	0.82	0.84	0.84	0.886
Recall	0.96	0.81	0.75	0.83	0.92	0.854	1.00	0.76	0.93	0.79	0.90	0.876
F1-score	0.94	0.83	0.78	0.86	0.87	0.856	1.00	0.84	0.87	0.82	0.87	0.880
River Dataset												
	Model without crop						**Model with crop**					
	3	**4**	**5**	**6**	**7**	**Mean**	**3**	**4**	**5**	**6**	**7**	**Mean**
Precision	0.00	0.25	0.09	0.75	0.50	0.318	1.00	0.29	0.28	0.77	0.60	0.588
Recall	0.00	1.00	0.29	0.20	0.10	0.318	0.50	0.33	0.71	0.67	0.30	0.502
F1-score	0.00	0.40	0.14	0.32	0,17	0.206	0.67	0.31	0.40	0.71	0.40	0.498



a) Complete Pool　　b) Cropped Pool

c) Complete River　　d) Cropped River

Fig. 6: Boxplots illustrating the predictions generated by the regression models (y-axis), categorized based on the ground-truth labels (x-axis). Ideally, the median of the predicted values should align closely with the low-variance labels. The predictions for the pool dataset were obtained using a stratified k-fold cross-validation approach, while the river dataset predictions were generated using the entire dataset (all models were solely trained on the pool dataset).

Table 2 presents the results of two distinct approaches: classification and regression problems. In each approach, models were trained and tested using both the complete images dataset and the modified dataset, in which images were cropped around the barcode panel. Various error metrics were utilized to assess the performance of these models.

To enhance the interpretation of the predicted results, confusion matrices were used, as shown in Fig. 7. Since the output produced by the regression models is continuous, we discretized the results by rounding them to the nearest class label of the classification model before constructing the confusion matrices.

5 Discussion

The results of ResNet V2 in the classification task in the pool dataset, as shown in Table 1, were quite similar in all metrics for pool dataset, which was observed across all classes and on average. The results were close to 85% - 88%. On the other hand, the regression model, as indicated by Fig. 6, had few outliers in

Table 2: Performance metrics for each proposed case. We evaluated the Mean Absolute Error (MAE), the Mean Squared Error (MSE), the Root Mean Squared Error (RMSE), and Coefficient of Determination R^2.

Pool Dataset				
	Classification		**Regression**	
	Complete	**Cropped**	**Complete**	**Cropped**
MAE	0.21	0.14	0.42	0.38
MSE	0.40	0.17	0.29	0.23
RMSE	0.63	0.41	0.54	0.48
$\mathbf{R^2}$	0.80	0.91	0.77	0.85
River Dataset				
	Classification		**Regression**	
	Complete	**Cropped**	**Complete**	**Cropped**
MAE	1.19	0.46	1.2	0.47
MSE	2.37	0.49	2.44	0.38
RMSE	1.54	0.70	1.56	0.62
$\mathbf{R^2}$	-20.12	0.48	-16.90	0.42

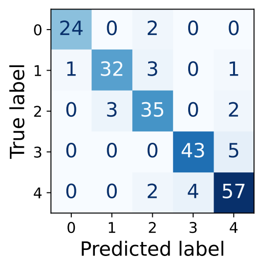
both cases. It should be noted that despite using the same images, the boxplots show that the predictions differed across the classes, which is also reflected in the outliers.

For the complete images, outliers occurred in classes 5 and 6, while for the cropped images, outliers occurred in class 7. Such factors are relevant to the problem presented, as these outliers would not negatively impact real-time detection and could be discarded.

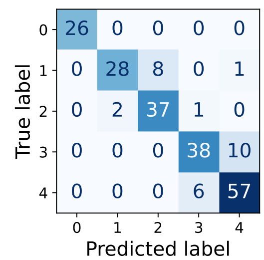
In the case of the river dataset, the boxplot for the complete images indicated poor results, with predictions mostly concentrated above 6. This outcome was expected due to the substantial environmental differences surrounding the barcode panel. However, for the cropped images, the distribution improved, with only one outlier detected in class 7. This finding suggests that utilizing the barcode panel can yield generalizable results when coupled with an object detection model that counts the bars. In terms of metrics, the results were inferior in this case, with the complete images achieving approximately 20-30% accuracy, while the cropped images achieved 50-59

In the pool dataset, confusion matrices showed that the classification models had few false positives, most of which were in neighboring classes. Noticeably, as the number of bars above the water increases, more false positives occur. However, for the regression models, the number of false positives was higher, but all false positives occurred in adjacent labels. This was probably due to the rounding of values.

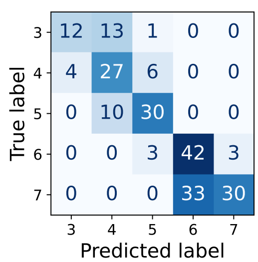
Using a regression model offers the advantage of providing continuous predictions for the water level, which can be more informative in certain cases. For example, a sequence of values like 5.1 followed by 5.4 would indicate an increasing water level. In the corresponding classification model, these values would both be rounded to label 5, thereby losing the nuance of the incremental change.



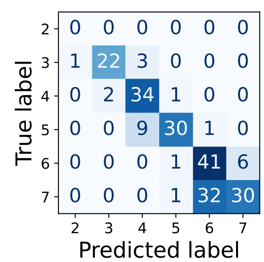
a) Complete classification Pool



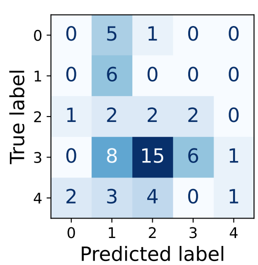
b) Cropped classification Pool



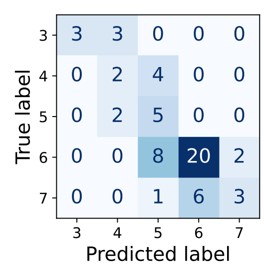
c) Complete regression Pool



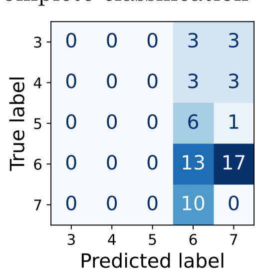
d) Cropped regression Pool



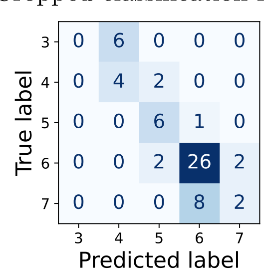
e) Complete classification river



f) Cropped classification river



g) Complete regression river



h) Cropped regression river

Fig. 7: Confusion matrices for all conditions analyzed.

Moreover, the confusion matrices for the river dataset reveal that complete images have a higher occurrence of outliers compared to the cropped images. In certain classes, the number of outliers in the predictions exceeded the number of correct predictions. However, similar to the pool dataset, the cropped images in the river dataset also exhibit outliers that correspond to adjacent labels.

In order to analyze the performance of the models, Table 2 showed that, in the pool dataset, MAE in regression models were two times higher than in classification, although both values were reasonably small. For RMSE, the complete classification model was worse than the others, and the models with crop were slightly better in this metric. Meanwhile, for R^2, the value was higher than 77% in all cases, reaching 91% in the classification model with crop.

The models with crop adapted better to the data. Compared to the complete images in the river dataset, all metrics were significantly better for the cropped images. However, the errors were larger in the river dataset, as expected since the models were trained on the pool dataset. The error results demonstrate that the models can be trained on one dataset and used to predict labels in another dataset, provided the images are cropped.

Furthermore, we highlight the performance of the models without cropping in the task of analyzing the barcode in the dataset we assembled. As illustrated in Fig. 3, the image exhibits exceptionally clear water, enabling the submerged bars to be distinctly recognizable. This aspect poses a greater challenge to the model and demonstrates its robustness. However, it is important to acknowledge that, in real-world applications, the water is expected to be more turbid, which can facilitate the model's performance.

6 Conclusion

This paper introduces a novel approach to measure water levels by employing a barcode panel in conjunction with deep learning models. Two distinct datasets were constructed: one in a pool, and the other one in an actual river. Machine learning models were trained using both regression and classification methodologies. Specifically, the ResNetV2 neural network architecture was employed in both approaches. To evaluate the potential improvements gained by integrating object detection techniques, the models were further trained and assessed using cropped images centered around the barcode panel.

The pool dataset yielded highly accurate results with minimal error rates. In this scenario, the cropped version demonstrated only a marginal improvement. However, in the case of the river dataset, the models trained with cropped images showcased better generalization capabilities compared to the models trained on complete images.

When contrasting the regression approaches with the classification models, the regression methods resulted in fewer outliers. However, there were slightly more false positives when considering the rounded outputs. Encouragingly, these false positives predominantly belonged to classes in close proximity to the expected class, minimizing the impact of misclassifications.

Future work will involve conducting studies with images not present in the datasets, such as images taken from other locations or rivers, as well as testing different lighting conditions. This will allow us to test the robustness of the proposed solution and make necessary adjustments. Additionally, we aim to build object detection neural networks to detect the barcode panel in the image and use it to auto-crop the images for performing the models. This auto-crop method can also be utilized to estimate the depth of the river.

References

1. Aljohani, F.H., Abi Sen, A.A., Ramazan, M.S., Alzahrani, B., Bahbouh, N.M.: A smart framework for managing natural disasters based on the iot and ml. Applied Sciences **13**(6), 3888 (2023)
2. Bioresita, F., Puissant, A., Stumpf, A., Malet, J.P.: A method for automatic and rapid mapping of water surfaces from sentinel-1 imagery. Remote Sensing **10**(2), 217 (2018)
3. Bolanos, S., Stiff, D., Brisco, B., Pietroniro, A.: Operational surface water detection and monitoring using radarsat 2. Remote Sensing **8**(4), 285 (2016)
4. Brazilian National Water and Basic Sanitation Agency - ANA: Conjuntura dos recursos hídricos no Brasil 2022: informe anual (2023)
5. Brito, L.A., Meneguette, R.I., De Grande, R.E., Ranieri, C.M., Ueyama, J.: Floras: urban flash-flood prediction using a multivariate model. Applied Intelligence pp. 1–19 (2022)
6. Ding, Y., Zhu, Y., Feng, J., Zhang, P., Cheng, Z.: Interpretable spatio-temporal attention lstm model for flood forecasting. Neurocomputing **403**, 348–359 (2020)
7. Do, H.N., Vo, M.T., Tran, V.S., Tan, P.V., Trinh, C.V.: An early flood detection system using mobile networks. In: 2015 international conference on advanced technologies for communications (ATC). pp. 599–603. IEEE (2015)
8. Fernandes Jr, F.E., Nonato, L.G., Ueyama, J.: A river flooding detection system based on deep learning and computer vision. Multimedia Tools and Applications **81**(28), 40231–40251 (2022)
9. Fernandes Junior, F.E., Nonato, L.G., Ranieri, C.M., Ueyama, J.: Memory-based pruning of deep neural networks for iot devices applied to flood detection. Sensors **21**(22), 7506 (2021)
10. Furquim, G., Neto, F., Pessin, G., Ueyama, J., de Albuquerque, J.P., Clara, M., Mendiondo, E.M., de Souza, V.C., de Souza, P., Dimitrova, D., et al.: Combining wireless sensor networks and machine learning for flash flood nowcasting. In: 2014 28th International Conference on Advanced Information Networking and Applications Workshops. pp. 67–72. IEEE (2014)
11. He, K., Zhang, X., Ren, S., Sun, J.: Identity mappings in deep residual networks. In: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14. pp. 630–645. Springer (2016)
12. Iqbal, U., Perez, P., Li, W., Barthelemy, J.: How computer vision can facilitate flood management: A systematic review. International Journal of Disaster Risk Reduction **53**, 102030 (2021)
13. Koyama, N., Sakai, M., Yamada, T.: Study on a water-level-forecast method based on a time series analysis of urban river basins—a case study of shibuya river basin in tokyo. Water **15**(1), 161 (2023)

14. Lei, X., Chen, W., Panahi, M., Falah, F., Rahmati, O., Uuemaa, E., Kalantari, Z., Ferreira, C.S.S., Rezaie, F., Tiefenbacher, J.P., et al.: Urban flood modeling using deep-learning approaches in seoul, south korea. Journal of Hydrology **601**, 126684 (2021)
15. Lin, Y.T., Lin, Y.C., Han, J.Y.: Automatic water-level detection using single-camera images with varied poses. Measurement **127**, 167–174 (2018)
16. Marengo, J., Alcantara, E., Cunha, A., Seluchi, M., Nobre, C., Dolif, G., Goncalves, D., Dias, M.A., Cuartas, L., Bender, F., et al.: Flash floods and landslides in the city of recife, northeast brazil after heavy rain on may 25–28, 2022: Causes, impacts, and disaster preparedness. Weather and Climate Extremes p. 100545 (2023)
17. Marengo, J.A., Seluchi, M.E., Cunha, A.P., Cuartas, L.A., Goncalves, D., Sperling, V.B., Ramos, A.M., Dolif, G., Saito, S., Bender, F., et al.: Heavy rainfall associated with floods in southeastern brazil in november–december 2021. Natural Hazards pp. 1–28 (2023)
18. Natividad, J., Mendez, J.: Flood monitoring and early warning system using ultrasonic sensor. In: IOP conference series: materials science and engineering. vol. 325, p. 012020. IOP Publishing (2018)
19. Pan, J., Yin, Y., Xiong, J., Luo, W., Gui, G., Sari, H.: Deep learning-based unmanned surveillance systems for observing water levels. Ieee Access **6**, 73561–73571 (2018)
20. Park, J.C., Kim, D.G., Yang, J.R., Kang, K.S.: Transformer-based flood detection using multiclass segmentation. In: 2023 IEEE International Conference on Big Data and Smart Computing (BigComp). pp. 291–292. IEEE (2023)
21. Patro, S., Chatterjee, C., Mohanty, S., Singh, R., Raghuwanshi, N.: Flood inundation modeling using mike flood and remote sensing data. Journal of the Indian Society of Remote Sensing **37**, 107–118 (2009)
22. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International journal of computer vision **115**, 211–252 (2015)
23. Sabbatini, L., Palma, L., Belli, A., Sini, F., Pierleoni, P.: A computer vision system for staff gauge in river flood monitoring. Inventions **6**(4), 79 (2021)
24. Strobl, B., Etter, S., van Meerveld, I., Seibert, J.: Accuracy of crowdsourced streamflow and stream level class estimates. Hydrological Sciences Journal **65**(5), 823–841 (2020)
25. Vandaele, R., Dance, S.L., Ojha, V.: Deep learning for automated river-level monitoring through river-camera images: an approach based on water segmentation and transfer learning. Hydrology and Earth System Sciences **25**(8), 4435–4453 (2021)
26. Wilfried, D.P., Rai, N., Singla, K.: Automatic flood detection by leveraging deep convolutional neural networks. In: 2023 11th International Conference on Internet of Everything, Microwave Engineering, Communication and Networks (IEMECON). pp. 1–6. IEEE (2023)
27. Wu, X., Zhang, Z., Xiong, S., Zhang, W., Tang, J., Li, Z., An, B., Li, R.: A near-real-time flood detection method based on deep learning and sar images. Remote Sensing **15**(8), 2046 (2023)
28. Yang, P., Ng, T.L.: Gauging through the crowd: A crowd-sourcing approach to urban rainfall measurement and storm water modeling implications. Water Resources Research **53**(11), 9462–9478 (2017)
29. Zhang, Z., Zhou, Y., Liu, H., Zhang, L., Wang, H.: Visual measurement of water level under complex illumination conditions. Sensors **19**(19), 4141 (2019)