**Flood Risk Management in Pakistan: The Role of Mask R-CNN in Flood Level Estimation**

Aasma Hameed1, Muhammad Aleem Raza 1, Usman Ahmed Raza2,

1Faculty of Computer Science & Information Technology, Lahore leads university, Lahore 54000, Pakistan

2 Department of Computational Intelligence, University of Naples, Federico II, Italy

Correspondence: Usman Ahmed Raza (Usmanahmed.raza@unina.it)

**Keyword**: Deep Learning, CNN, Mask R-CCN, Flood level Estimation, Social media Image Recognition

**Abstract**

According to the National Disaster Management Authority (NDMA), one of Pakistan's worst floods, triggered by exceptionally heavy monsoon rains, affected around 33 million people nationwide. During floods, it’s necessary to gather observations from the disaster site, and these observations are used to create flood-level maps to aid in emergency operations. People in flooded areas shared text and images on Social media platforms to show the current situation. In this paper, we suggested a mask R-CNN model-based approach to predict flood water levels on images collected from social media. We face these challenges i) the size of the object that appears in the image is unknown ii) There may be variations in the height of the flood water that appears in various parts of the image, and iii) the objects that may be submerged in water are partially visible. We addressed these issues by identifying class objects with known sizes to estimate water levels. We train the Mask R-CNN model on the flood-water dataset and then finally check the ability of the train model on collected images from real-time flood areas. The Mask-RCNN model achieved 0.85 accuracy in detecting submerged objects with an error margin of just 0.15 cm in water level estimation.

1. **Introduction**

Due to the impacts of global climate change, floods are expected to increase in both frequency and severity, leading to greater potential damage [14]. Flooding is a significant environmental challenge in Pakistan, which has experienced numerous floods from 1973 to 2022, with some of the most severe occurrences in recent years. In August 2022, Pakistan experienced one of the most devastating floods in its history, driven by exceptionally heavy monsoon rains that impacted around 33 million people nationwide. The National Disaster Management Authority (NDMA) reported that these floods led to 211 fatalities, 294 injuries, the destruction of nearly 4,000 homes, and the loss of over 1,100 livestock additionally, agricultural losses in the most productive Indus plains have exacerbated the risk of food insecurity in the country [15].

To decrease human death, accurate maps of flood areas are essential for effectively planning rescue operations. Real-time, scattered data about flood water levels from the disaster area must be obtained to create such maps. There are several drawbacks to routine monitoring methods including field data gathering, remote sensing, and stream gauging. For example, satellite-based remote sensing data is very cheap, does not provide real-time data during a crisis, and usually has an excessively large revisit cycle [4]. The data collection on the field is usually too costly and alarming as it needs to examine the area of disaster. This time, the social media platform offers a useful substitute source of information. Flood-affected areas of People often share images describing the situation. The most significant advantage of this data is that it is readily available in real-time, directly from the flooded region, and it is also quite inexpensive. The main drawback is that to extract useful information from the image, it must be correctly processed. No one method has been proposed yet that retrive automatically floods information from social media images.

We aim to close this gap in this research by introducing proposed a deep learning framework, Mask R-CNN for flood-water level estimation based on images. Estimating flood levels from the image is a challenging task. The primary challenge arises from the potential instability of water levels, as they can vary across different areas captured within an image. To address this issue, we analyze each distinct object in the image individually, estimating the extent to which each is submerged in water. If we know the approximate knowledge of the different objects' heights, the water level can then be roughly estimated. However, examining each object separately uncovers more difficulties: i) the objects are partially visible or submerged in water; ii) it is impossible to determine the exact size of the objects. Recognizing objects, even when they are only partially visible, is crucial, as they provide valuable information about the water level. Fortunately, deep learning models, like the one we employ, can detect objects even when they are only partially visible. To overcome the second challenge, we search for certain things that are similar in both, making them likely to show up in the images, and whose size is approximately consistent throughout all occurrences. This approach allows us to determine the water level relative to each object and make a comprehensive overall estimate.

The main goal of this paper is to contribute to the development of such a system by showing how image-based sensing and detection techniques could be utilized to detect flood levels based on the object present in the image. In this paper, we proposed the Mask R-CNN model for the estimation of flood water levels. Mask R-CNN is built on a Faster R-CNN algorithm and improves the segmentation performance. The implementation of Mask R-CNN on detecting floodwater has not been done previously. Next, we estimate the amount of water submerged in the objects that fall into particular categories using the Faster-RCNN algorithm once more. Once the network architecture has been established, we create a dataset by adding flood-water level information to images, which we then use to train our network. Next, we assess the suggested model to demonstrate its efficacy in estimating the water level of objects in images.

The remainder of this paper is organized as follows: Section 2 presents a review of existing research on water level estimation and flood prediction technologies. Section 3 details the methodology, including data acquisition, model selection, and implementation. In Section 4, we discuss our experimental results and evaluate the model's performance. Finally, Section 5 concludes with a discussion of the implications of our findings, as well as potential directions for future research to further enhance water level estimation capabilities.

1. **Related Work**
   1. **Flood Level Estimation**

In this study, a deep learning model was created to forecast the flood water level based on images collected from social media. This model was created on behalf of the Mask R-CNN Architecture. In the feature, they expanded her framework and also used text information to gather flood data. In the feature, they also plan to investigate the more advanced method where they combine water level prediction of each object instance [8].

In this paper, they developed an efficient ResNet-50 model to predict the human pose that information gathered from the social media image. They used multimodal flood level estimation and also they achieved the best Accuracy score 95.6%.In the feature, they plan to increase the dataset to improve the accuracy of the restnet50 model, and this framework is implemented as the practical system [7].

In this paper, they present her role in the medieval 2019 task to investigate the flood water level from news images. They combine both methods such as pose detection and level estimation those objects that stand in the water. The limitation of this paper is that they only estimated the flood water level of humans and no other object present in the water [3].

They presented an improved Mask R-CNN model in this research. They improved some factors of Mask R-CNN that led to a productive network. They enhanced the feature extractor of the Mask R -CNN model that enhancement made the smaller and faster model. They compared this optimized model with another model to check the speed characterized and they showed better results. The four categories of electron components are detected and segmented using this model. To improve the robustness of the revised model, the feature will include more electronic components and a larger dataset [2].

In this paper, they Mask R-cnn model for the detection of tea picking point. They selected only 100 images of tree bud and leaf pictures for testing purposes. Mask R CNN shows an overall accuracy rate is 93.95% and a recall rate is 92.48%. Mask R-CNN model results compared with other methods but the mask R-CNN shows better results identification of positioning during complex environments [1].

* 1. **Mask-RCNN Object detection**

The method, called Mask R-CNN, extends Faster R-CNN by incorporating a branch for object mask prediction with the existing branch for bounding box identification. Mask R-CNN only adds a small overhead, whereas Faster R-CNN operates at 5 frames per second. Additionally, Mask R-CNN is easy to adapt to other applications, allowing us to estimate human poses, for instance, inside the same framework. They achieve the highest scores in the instance segmentation, bounding-box object detection, and person key point detection challenges from the COCO suite. On every job, including the COCO 2016 challenge winners, Mask R-CNN surpasses all currently available single-model entries without the use of any gimmicks [6].

1. **Materials and Methods**
   1. **Dataset**

In this paper we have used two datasets, the first dataset contained the flood image collected from social media from Pakistan, and the second dataset that has been used is the MS COCO dataset. We also give a brief overview of the dataset like flood image collection and flood image annotation strategy and generating the flood image masks.

* + 1. **Flood dataset**

In this paper, we evaluate flood water level on the base of the object that submerged in flood water, during the data collection we take first is which object we are going to consider for level detection. During selecting an object we define criteria for data collection our criteria are based on the easy availability know dimensions and less variation in the height of the sub-classes. When the availability of the object is high enough frequency we can collect a large number of the dataset. So when we know about the dimensions of the object it refers to the fact that height length and width can be approximated and have less height difference, for example, if we consider a car as an object, the cars come with different types of category and should not vary in height from each other [2].

In this paper, our main concern is about the height of objects. Selected five classes of objects such as Persons, cars, buses, bicycles, houses, or Buildings based on the criteria we decided. So when we selected images for our flood dataset at least one of these five objects was in them. After removing the duplicated images we have only 104 images in the flood dataset. Flood water images were only collected from the National Disaster Management Authority (NDMA) Pakistan and social media platforms [16].

* + 1. **MS COCO dataset**

A sizable dataset for object detection, segmentation, and captioning is MS COCO. It was created by Lin et al. [Lin+14] and released in 2015 [2]. The dataset is sponsored by Mighty Ai, Facebook, Microsoft, and CVDF. It has 91 categories of things and 80 categories of objects, over 200 thousand tagged photos, and 1.5 million instances of objects. The dataset comprises an average of 7.7 instances and 3.5 categories per image. The dataset comes in two versions, one from 2014 and the other from 2017 [Con18]. We used the 2017 version for this job, which contains 118,000 train images and 5000 validation images [Con18]. We just use a portion of the object categories namely, Person, Car, Bus, and Bicycle because we don't require all of them for our work. The entire list of object categories is included in Appendix A.2. By giving the object category ID at runtime, the task of choosing a subset of photos is accomplished, and only images that include this category ID are chosen for the training and validation set [2].

* 1. **Image Annotation**

We need to label the flood images in our training dataset with the flood-water level we want to predict to train our Mask R-CNN model. We first established the annotation method and chose which objects to take into account for the classification task to achieve our goal of quantifying flood water level depending on the object that is flowing in the water. The objects for this task were selected based on the following criteria: known dimensions, convenient availability, and little intra-class height variance. Because the object is widely available in the real world, it was simple to gather a sizable collection of images including the object for training and validation [2]. We mean known size describes the height and width of the considered object that is known, it’s simple for level estimation. In the end, we considered the low variations of the height of the object which means many of the objects are around the same height in the actual world. For example, Bicycle objects are hugely common in rural areas and are also known for their size, and across different model their height is constant. Based on the criteria we decided to select five object classes for our research: person, car, bus, bicycle, and house.

We chose the flood class that describes the flood water in the image along with to all five classes that have already been discussed. In this research, we investigate 11 flood levels, ranging from 0 (no water) to 10 (the height of a person buried in the water). Creating the training dataset is the next stage of our research, and to do so, we must annotate the images with information on the water level. The Visual Geometry Group developed the open-source image annotation program VGG Image Annotator (VIA), which we used. We collected a total of 500 images during the labeling process we only labeled 130 images for training and validation due to image quality.

In this paper, we used VGG Image Annotator (VIA), to label the image's shape region, and that region was set to object submerged in water. The height of each level is then inspired by artists who sketch the human figure using head height as the basic measure. To map the level classes to actual flood height, we take the average height of the human body into account and obtain the water height in centimeters (see Table 1).

|  |  |  |  |
| --- | --- | --- | --- |
| **Level Name** | **Range cm** | **Value nearest integer cm** | **Feet** |
| Level-0 | No water | 0.0 | 0 |
| Level-1 | 0.0–1.0 | 1.0 | 0.03 |
| Level-2 | 1.0–10.0 | 10.0 | 0.3 |
| Level-3 | 10.0–21.25 | 21.0 | 0.6 |
| Level-4 | 21.25–42.5 | 43.0 | 1.4 |
| Level-5 | 42.5–63.75 | 64.0 | 2.1 |
| Level-6 | 63.75–85 | 85.0 | 2.9 |
| Level-7 | 85.0–106.25 | 106.0 | 3.4 |
| Level-8 | 106.25–127.5 | 128.0 | 4.1 |
| Level-9 | 127.5–148.75 | 149.0 | 5.0 |
| Level-10 | 148.75–170.0 | 170.0 | 5.5 |

Table 1: Water-level to centimeters relations table

In this paper, we can improve the annotation strategy, by considering the five classes of the objects as the average height. We have estimated the average height of the objects in the real world.

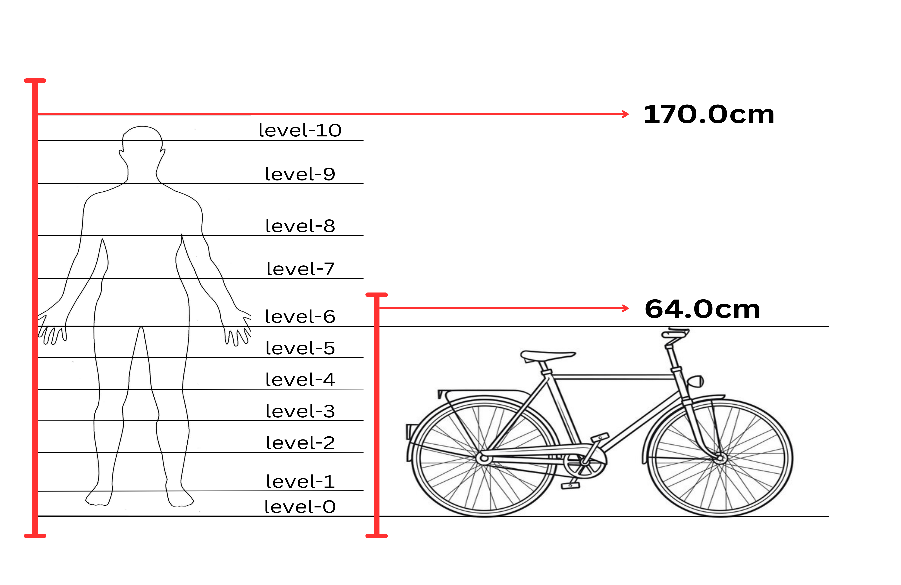


Figure 1: Annotation techniques for object and person and bicycle

* 1. **Model training and loss Function**

For the experimental study, we utilized a pre-trained Mask R-CNN model. These experiments were performed on a personal laptop equipped with Intel Intel(R) Core(TM) i5-8365U CPU @ 3.40GHz and Intel(R) UHD Graphics 620 through Anaconda Jupyter notebook and Kaggle notebook also used. In this paper, we propose a deep-learning approach for flood-water level estimation. We used pertained Mask R-CNN as the base architecture which is the best and easiest solution for instance image segmentation. Figure 2 shows the overall diagram of the method.

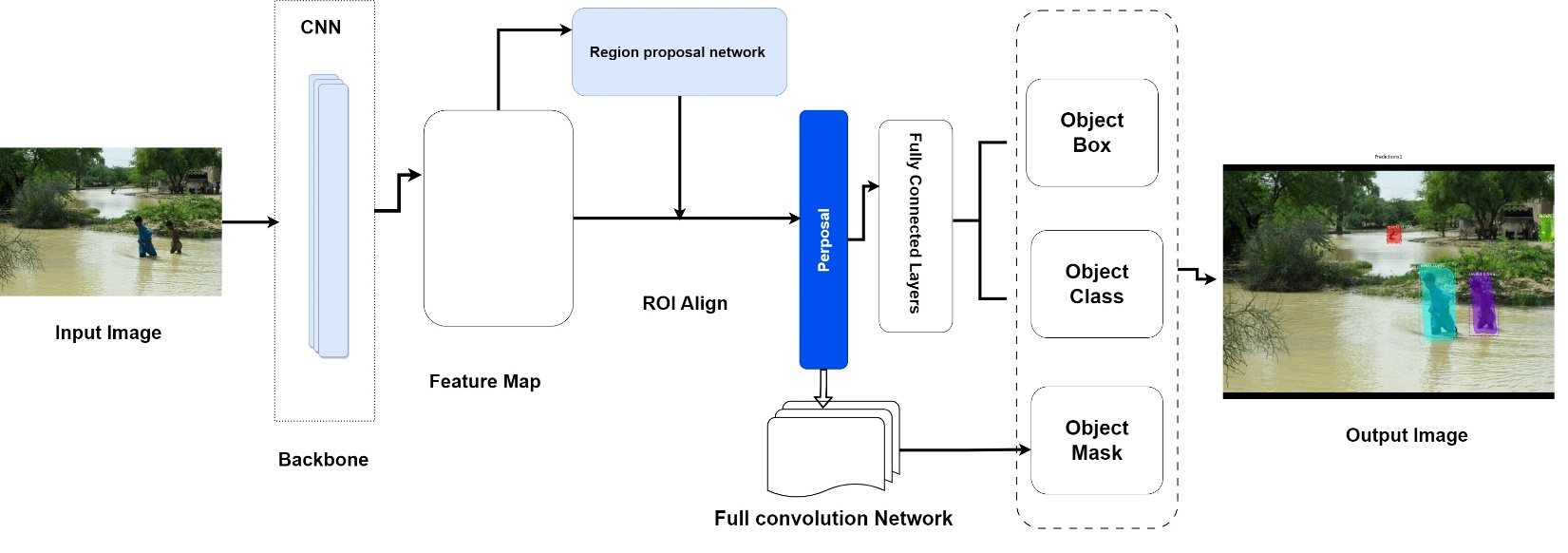


Figure 2: Architecture of Mask R-CNN for Water-level estimation image segmentation

For the segmentation process, there are steps required, (i) level Detection (ii) level classification (iii) object localization. The class label predicted for each level is coupled with a bounding box of the images with (X, Y) coordinates in the Level Detection method. Every input image was divided during the segmentation process, and each pixel value was predicted and assigned a class level. The deep learning model mask R-CNN was implemented in this research using an algorithm that was published by He et al. in 2017. This method also enhanced the earlier mask R-CNN, Fast R-CNN [39], and Faster R-CNN algorithms [40]. The Faster R-CNN model uses the predicted class labels and bounding box, and two convolutional layers that are used to predict the mask of each detected object and show the visualization of the object location. Using Keras and TensorFlow, our mask R-CNN network [41] loads a pre-trained ResNet101 network from the ImageNet database. Using our training dataset of 196 photos and our validation dataset of 30 images, ResNet101, with its 101 layers, was able to extract useful features and provide a high degree of gland detection and classification accuracy. The anchor box scales (8, 16, 32, 64, and 128) and anchor ratios [0.5, 1, and 2] are adopted by the RPN (region proposal network). The square anchor sides' lengths in pixels are 8, 16, 32, 64, and 128. This means that for each anchor point, five square anchor boxes were made, each measuring 8 × 8, 16 × 16, 32 × 32, 64 × 64, and 128 × 128 pixels. Using the three anchor ratios, five × three anchor boxes were made for each anchor point. The greatest results for our flood level detection goal would be obtained by combining anchor scale and anchor ratios. Anchor scales ought to be adjusted in accordance with the general shapes of the objects that have been detected. To ensure that the mask had the same dimensions as the original image, we used the whole mask width during the training phase. During the training, we used another method called mini mask is generated by the bounding box of the object and resized on a pre-defined shape. For example, a full-size mask of the image is 1024 x 1024 pixels and resized to a size of 224 x 224 pixels and it’s very helpful for memory requirements during the training process. 5.0 sigma. The optimizer utilizes ADAM in conjunction with root mean square propagation (RMSProp) and adaptive gradient algorithm (AdaGrad). When there are sparse gradients, AdaGrad performs better, but the RMSProp algorithm performs best when there is noise. The learning momentum was 0.9, the learning rate was 0.001, and the weight decay was 0.0001. A considerable amount of image augmentation was used, which included flipping 50% of all shots vertically and 50% of all photos horizontally, applying Gaussian blur with 0.0–5.0 sigma, and applying affine rotation by 90, 180, and 270 degrees. The architecture of the mask R-CNN network. Image scanning and regional proposal generation for possible objects are the first two steps of the mask R-CNN (regional convolutional neural network) framework. Proposal categorization, pixel-wise mask generation, and bounding box generation come next. This particular network uses FPN for feature extraction in addition to ResNet101 as its foundation [2].

Backbone feature maps are scanned by the region proposal network, or RPN, which removes pointless calculations and permits the reuse of features that have been extracted. The RPN generates two outputs for each anchor: anchor class (foreground or background: foreground signals the likely existence of an item) and bounding box refinement (the foreground anchor box is resized and positioned to better match the object).

The last round of proposals moves on to the next phase. As of right now, the RPN generates two outputs for each ROI: bounding box refinement and class (for objects). An ROI pooling technique enables classifier capability by cutting a portion of a feature map and enlarging it to a specified size. At this point, two convoluted layers are constructed simultaneously to create a branch that creates masks for the ROI classifier's chosen positive regions. The backbone of Box Class Mask Input RoI Pooling for RPN Feature Map Figure 2: Fully Connected Layers Convolution Network. Using the ROI pooling outputs, the other branch of fully connected layers produces two values for each object: a class label and a bounding box prediction.

* 1. **Mask Generation and loss function**

In the mask generation phase of Mask R-CNN, a compact neural network processes the detections from the final layer to produce masks for each identified object instance. To maintain efficiency, these masks are initially generated at a smaller scale and then appropriately resized to fit the detected objects.

The overarching architecture builds upon the foundation of Mask R-CNN, with the total loss comprising four primary components: classification. Bounding box regression flood level classification, and mask prediction

(1)

* 1. **Mask Prediction Loss**

The mask prediction loss is computed using binary cross-entropy, penalizing deviations between predicted and ground truth Mask values across each mask cell (𝑖, *i*,*j*) and class (𝑘*k*). It operates over the entire mask grid (𝑚×*m*×*m*) for each detected object instance.

(2)

Lmask: This is the mask prediction loss. 𝑚m: Represents the size of the mask grid (typically 𝑚×𝑚m×m). 𝑦𝑖𝑗yij: Ground truth value of the mask cell at coordinates (𝑖,)(i,j). 𝑦^𝑖𝑗y^ij: The predicted value of the mask cell at coordinates (𝑖,)(i,j). Log: Natural logarithm. This equation computes the binary cross-entropy loss for each mask cell and each class. It penalizes discrepancies between the predicted mask values and the ground truth mask values.

* 1. **Classification Loss**

log ()] (3)

For classification *L*class​ and flood level 𝐿level tasks, cross-entropy loss measures are employed, assessing the accuracy of class predictions across all classes’ 𝐾 and flood levels 𝐿, respectively.

𝐿class: This is the classification loss.𝑁class: Total number of classes. 𝑦𝑖𝑘: Binary variable indicating whether anchor 𝑖i belongs to class 𝑘k. 𝑝𝑖𝑘: Score of anchor 𝑖i being classified into class 𝑘. Log: Natural logarithm. This equation computes the cross-entropy loss for classification, penalizing incorrect class predictions.

* 1. **Flood Level Classification Loss**

log ()] (4)

𝐿level: This is the flood level classification loss.𝑥𝑖𝑙: Binary variable indicating whether anchor 𝑖 belongs to flood level 𝑙.𝑞𝑖𝑙: Score of anchor 𝑖 being classified into flood level 𝑙.log: Natural logarithm. This equation computes the cross-entropy loss for flood level classification, penalizing incorrect level predictions.

* 1. **Bounding Box Regression Loss**

Bounding box regression 𝐿bbox employs a smooth L1 loss function, addressing discrepancies between predicted (𝑡𝑖*ti*​) and ground truth (𝑡𝑖∗*ti*∗​) bounding box coordinates. It spans all bounding boxes and is weighted by a balancing parameter (𝜆*λ*) to modulate its impact relative to other losses.

()] (5)

𝐿bbox: This is the bounding box regression loss.𝜆: Balancing parameter.𝑁bbox: Normalization term.𝑝𝑖∗: Binary variable indicating whether anchor 𝑖 is an object.𝑡: Predicted bounding box coordinates for anchor 𝑖.𝑡𝑖: Ground truth bounding box coordinates for anchor 𝑖.𝐿smooth1: Smooth L1 loss function. This equation computes the regression loss for bounding box coordinates, penalizing discrepancies between predicted and ground truth coordinates, weighted by 𝜆.

1. **Experimental Results and Analysis:**

These experiments on the deep learning Mask R-CNN model were to the estimation of flood level. The Mask R-CNN model was trained on the 70 epochs. We used 1 GPU for two images and trained a model on the 11 classes starting from a learning rate of 0.001. In this configuration, training on a single GPU took four hours. Table 1 tabulates the Mask R-CNN model's performance.

* 1. **Evaluation Metrics:**

In this paper the evaluation of the Mask R-CNN model on the base of four common metrics which are detailed below.

* 1. **Accuracy**

The percentage of accurate predictions produced by the Mask R-CNN model across all classes is measured using the accuracy metric. By dividing the number of accurate forecasts by the total number of predictions, the accuracy was determined. The range of the accuracy is between 0 to 1, when the accuracy is 1 the show the perfect prediction of the accuracy. The accuracy was calculated using the following formula.

(6)

Where the numbers for true positives, true negatives, false positives, and false negatives are represented by the TP, TN, FP, and FN, respectively.

* 1. **Precision**

The ratio of true positive (TP) predictions (that is, accurately predicted positive instances) to the total of TP and FP was measured by the precision matrix. The model has a low false-positive rate when the accuracy score is 1. The following formula was used to determine the precision.

*Precision* = (7)

Where the numbers for True Positive and False Positive are denoted by TP and FP, respectively.

* 1. **Recall**

Recall, often referred to as sensitivity, was computed using true positive predictions as a percentage of all real positive instances; a low false negative rate was indicated when the recall score was near 1 significance. The following formula was used to determine the recall.

*Recall* = (8)

Where TP and FP are the numbers of true positives and false negatives respectively.

* 1. **Mask R-CNN Model Training and Validations loss Curve**

The above Figure 3 shows the training and validation loss curves for a Mask R-CNN model implemented for water level estimation. In Figure 3 each subplot explained the performance of different components of the Mask R-CNN model on 70 epochs. In the figure, over loss curve shows a steady decrease in training loss, which shows effective learning. Nevertheless, the validation loss, following an initial decline, reaches an endpoint and experiences a tiny increase, indicating the possibility of overfitting. This tendency is a frequent indicator that the model is remembering the training data instead of accurately making inferences from new data. The Region Proposal Network (RPN) classification loss plot shows that the training loss reduces substantially and becomes stable quickly, however, the validation loss swings and does not continuously decline. This fluctuation suggests that the model's ability to discriminate between object and non-object areas is unreliable with further information. Likewise, the RPN box loss curve indicates a gradual reduction in training loss, but the validation loss stays large and varied, indicating the difficulty in reliably anticipating bounding boxes for unknown data. Both "the Mask R-CNN classification and box loss curves" indicate diminishing training loss trends, indicating that the model's predicted class and predicted bounding box for identified objects are becoming more accurate. However, the validation losses for such variables are more unpredictable and do not show the same continuous decline, implying overfitting or probable noise in the validation data. Consequently, a distinct declining trend in training loss is shown in the mask loss plot, which is essential for accurate division like water level estimation. However, the validation loss shows variations and does not follow an even curve, indicating that the model's segmentation performance on fresh data might be improved. As a whole, while the training curves show excellent learning, the unpredictability, and larger values in the validation losses indicate that the model might benefit from improved generalization Procedures. Regularization methods, data augmentation, and getting a more diverse and representative validation set might all be used to assure robust performance across many circumstances. By thoroughly examining these loss curves, we get significant insights into the Mask R-CNN model's training mechanics and pinpoint particular areas for further development [10]. Addressing these challenges will be critical to increasing the model's accuracy and dependability in predicting water levels and, as a result, more active monitoring and management of water resources.

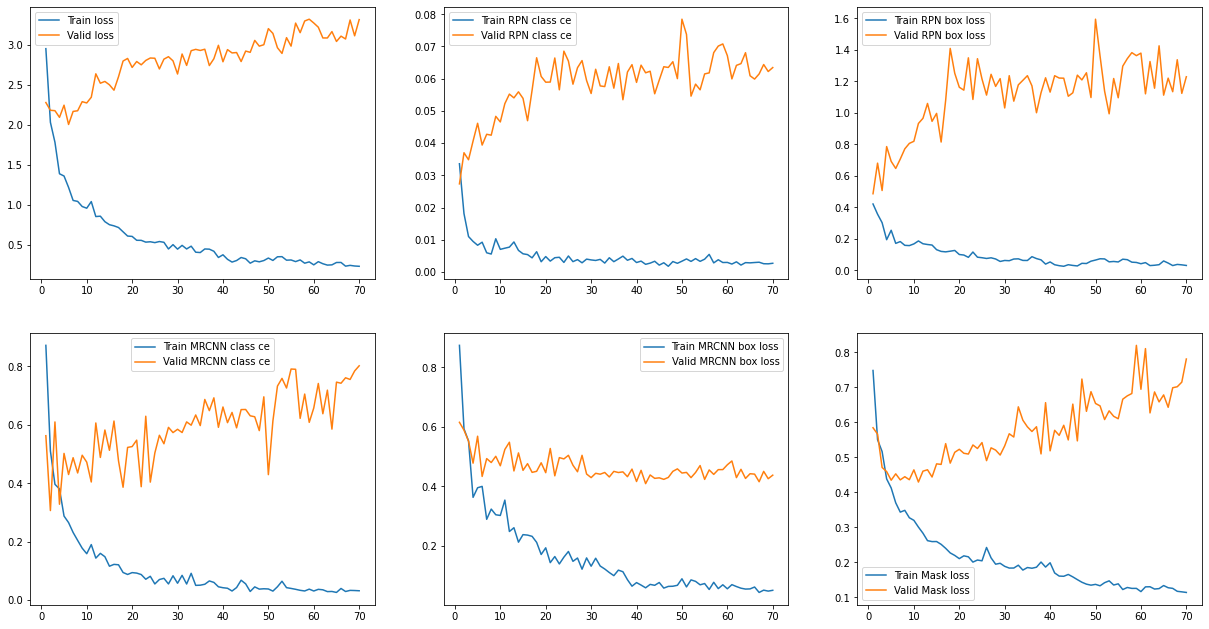
****

Figure 3: Training and validation loss of Mask R-CNN

* 1. **Precision/recall curves**

The precision-recall curve for the Mask R-CNN model used to water level estimate is shown in Fig 4 above, with an Average Precision (AP) at the IoU threshold of 0.50 (AP@50=0.500). This curve is critical when measuring the model's performance in terms of accuracy and recall at various threshold settings. When recall is the lowest point the precision-recall curve begins at 1.0, showing that the model delivers highly accurate predictions with few true positives. However, when recollection grows, accuracy decreases dramatically. This indicates that, while the model is capable of high precision, maintaining that precision becomes more difficult as it seeks to extract more true positives. The sharp decrease in accuracy points out that the model may struggle with false positives as the recall rate rises. An AP@50 of 0.500 shows that the model performs moderately in terms of precision and recall. This measure integrates accuracy and recall into a single value, which represents the area under the precision-recall curve. A result of 0.500 indicates that the model achieves a reasonable balance of accuracy and recall at the IoU threshold of 0.50, but there is still potential for enhancement. The significant dip in the curve indicates possible areas for improvement. Improving the model's capacity to sustain better precision at higher recall levels may require polishing the training data, improving the model architecture, or using sophisticated approaches such as hard negative mining. These techniques might assist achieve a more appropriate precision-recall bond, which is critical for accurate water level measurement. Ultimately, while the Mask R-CNN model performs rather well with an AP@50 of 0.500, more tweaks and optimizations are required to improve its precision-recall balance, thereby increasing the model's usefulness in real-world water level determine tasks.

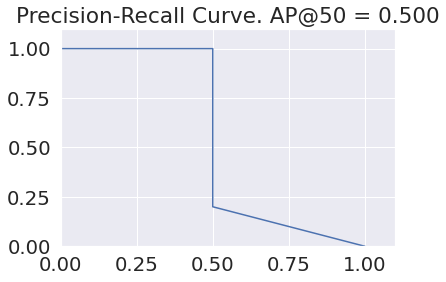


Figure 4: Show Precision and Recall of Mask R-CNN

* 1. **Confusion matrix**

The provided confusion matrix in Figure 5 shows an extensive description of the Mask R-CNN model's performance in water level estimates at various levels. Each cell in the matrix denotes the number of times the real water level (rows) was forecasted as a distinct water level (columns). The diagonal components represent accurate predictions, but the off-diagonal components represent misclassifications. Correct forecasts of diagonal components have the largest count, demonstrating the model accurately predicts water levels in a broad range of situations. However, there are considerable off-diagonal components, indicating that there is still opportunity for development at some levels. The background class (bg) contains several misclassifications, resulting in inaccurate predictions for multiple water levels. This shows that the model has trouble distinguishing between background and real water level classes. Levels 3, 5, and 9 exhibit enhanced performance with higher correct predictions. However, levels 0, 1, and 2 exhibit greater misclassification rates, indicating that the algorithm struggles to reliably anticipate these precise water levels. A brief review of the model's accuracy at each level is provided in the summary row at the bottom. Level 3's accuracy is 77.14%, whereas level 5's accuracy is 81.82%. These numbers imply that although the model works well at some levels, total accuracy differs at other levels. The model's performance may be enhanced in the areas indicated by the matrix. The model's overall performance would be improved, for example, by decreasing misclassifications in the background class and increasing accuracy at levels with lower correct prediction rates. We may learn more about the unique advantages and disadvantages of the Mask R-CNN model for water level estimate by examining the confusion matrix. This comprehensive review aids in revealing areas that require focused enhancements, such as improving feature extraction strategies, honing training data quality, or putting in place more advanced post-processing procedures. Enhancing these elements will be essential to obtain more precision and dependability in water level estimates, which will eventually help to improve water resource monitoring and management.

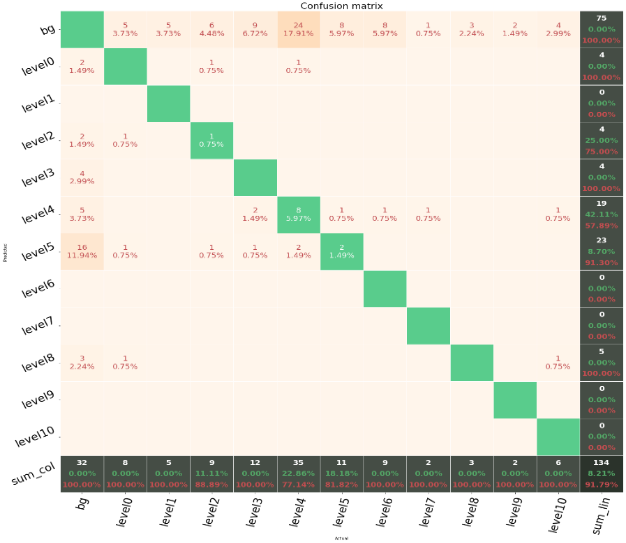


Figure 5: Confusion Matrix of our classes and level

1. **Qualitative Analysis of Mask R-CNN Model for Water Level Estimation**

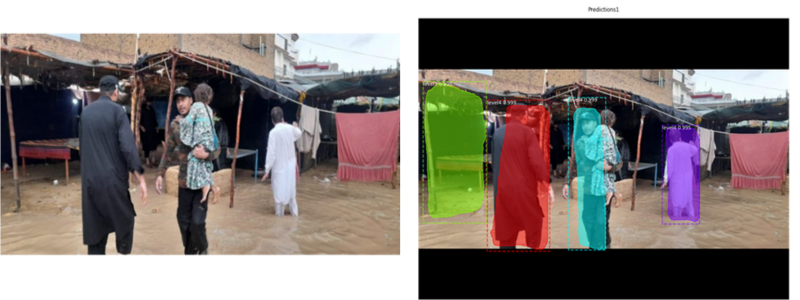
****

Figure 6: qualitative evaluation of test images.

Figure 6 is used for demonstrating the qualitative analysis of the Mask R-CNN model's performance on water level estimation: the original image and the detection result. The Mask R-CNN model successfully detects and classifies water level demonstrating its ability in real work scenarios. The Original Figure 6 shows a flood scene with individuals navigating through the water, and this figure is used to identify the performance and model detection accuracy. In the detection, the Mask R-CNN model efficiently detected and segmented the different regions with marked water level colored with bounding boxes and labels. The model correctly recognizes multiple individuals and allocates appropriate water level labels to each detected region. For example, the person on the left is accurately labeled as level 3 with a high confidence score of 0.998, while and person in the middle carrying a child is labeled as level 4 with a confidence score of 0.999. The model's strong certainty in its predictions is indicated by the high confidence scores for every detected region. This implies that the model has been properly trained to discriminate between various water levels. The segmentation masks show how effectively the model works for accurate localization and segmentation of water levels because they match the borders of the identified locations.

* 1. **People on Objects or Dry Surface**

****

Figure 7: qualitative evaluation of test images.

The Mask-RCNN model accurately detected the people walking through or standing in the water with high confidence level. Each object was emphasized with a bounding box and confidence levels, which shows our model was able the detect accurately. Our Mask-RCNN model had trouble with accurately detecting the objects that were standing on the dry surface. For example, some individuals in Figure 7 are standing on a dry surface or being carried, our model detection missed some aspects of the scene. Our model shows the worst results when distinguishing between the object on solid ground and the object, so also when the background of the figure is less uniform or more complex. The possible reason that object backgrounds are less uniform and more complex our model shows worse results and bad accuracy in detecting the objects are present on the dry surface. Another possible reason for our model's worst results, our model might be training on the data with more examples of objects in water, which makes it more adept at detecting those objects over the other.

* 1. **Houses or buildings are detected but one is not detected as well**

****

Figure 8: qualitative evaluation of test images.

The object in Figure 8 our train model successfully detected the house and person with a high confidence level, this indicated that our model is well-trained. On the objects, the bounding box accurately aligns, especially around the house, which shows the high segmentation ability of our model on this dataset. The water level that reaches the center and lower portions of the home has been annotated in the model, and this precisely matches the observable extent of flooding. The model's capacity to contextualize water height concerning other things is demonstrated by the marking of flood levels for each recognized human. This is essential for assessing flood risk since it aids in determining possible threats to public safety. The detection findings make it simple to visually evaluate the data by using different colors to distinguish items (red for the house, for example, and other colors for individuals). On the other hand, increasing the transparency of color overlays may make it easier to see the underlying image elements, like textures and waterline borders. The labels enhance the visual display with quantitative data by providing confidence scores and flood-level figures. The scores are easy to interpret and aid in determining each detection's reliability. Although the model recognizes and labels the house and people well, adding support for other things in future images, such as cars and bicycles, might increase the model's adaptability. Since there were no cars, buses, or bicycles in this image, it would be useful to test the model on images that have these types of items to assess the model's detection consistency across all targeted categories.

1. **Conclusion**

In this study, we addressed the significance of automatic disaster management systems as well as their difficulties, as specifically used to floods. Using a deep learning framework, we developed a model in this paper that uses images gathered from social media platforms to predict the flood water level, and this model is based on the Mask R-CNN architecture. When a specific object is detected, our suggested model not only fast segmentation but and same time the predicts water level. We plan to enhance our architecture to better utilize textual data in the feature. In actuality, images shared on social networking sites frequently have text attached that discusses the image's veracity. The realization is that we can increase the prediction's accuracy even further if we can integrate these two connected pieces of information.

1. **References:**

[1] Wang T, Zhang K, Zhang W, et al. Tea picking point detection and location based on Mask-RCNN. Inf Process Agric. 2021.

[2] Yang Z, Dong R, Xu H, Gu J. Instance segmentation method based on improved mask R-CNN for the stacked electronic components. Electronics. 2020;9(6):886.

[3] Chaudhary P, D'Aronco S, Moy de Vitry M, Leitão JP, Wegner JD. Flood-water level estimation from social media images. ISPRS Ann Photogramm Remote Sens Spatial Inf Sci. 2019;4(2/W5):5-12.

[4] Chaudhary P. Floodwater-estimation through semantic image interpretation. [dissertation]. Munich, Germany: Technical University Munich; 2018.

[5] Lin TY, Maire M, Belongie S, et al. Microsoft COCO: common objects in context. In: Computer Vision–ECCV 2014. 13th European Conference; September 6-12, 2014; Zurich, Switzerland. Springer International Publishing; 2014:740-755.

[6] He K, Gkioxari G, Dollár P, Girshick R. Mask R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision. 2017:2961-2969.

[7] Quan KAC, Nguyen VT, Nguyen TC, Nguyen TV, Tran MT. Flood level prediction via human pose estimation from social media images. In: Proceedings of the 2020 International Conference on Multimedia Retrieval. 2020:479-485

[8] Zhang JK, Fanous M, Sobh N, Kajdacsy-Balla A, Popescu G. Automatic colorectal cancer screening using deep learning in spatial light interference microscopy data. Cells. 2022;11(4):716.

[9] Podder S, Bhattacharjee S, Roy A. An efficient method of detection of COVID-19 using Mask R-CNN on chest X-ray images. AIMS Biophys. 2021;8(3):281-290.

[10] Farahnakian F, Sheikh J, Farahnakian F, Heikkonen J. A comparative study of state-of-the-art deep learning architectures for rice grain classification. J Agric Food Res. 2024;15:100890.

[11] Chaudhary P, D’Aronco S, Leitão JP, Schindler K, Wegner JD. Water level prediction from social media images with a multi-task ranking approach. ISPRS J Photogramm Remote Sens. 2020;167:252-262.

[12] Fujita H, Itagaki M, Ichikawa K, et al. Fine-tuned pre-trained Mask R-CNN models for surface object detection. arXiv Preprint. Published October 2020. Accessed [date]. doi:10.48550/arXiv.2010.11464

[13] Ren S, He K, Girshick R, Sun J. Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Trans Pattern Anal Mach Intell. 2016;39(6):1137-1149.

[14] Liang L, Huang W, Awan M, et al. Study and application of image water level recognition calculation method based on Mask RCNN and Faster R-CNN. Appl Ecol Environ Res. 2023;21(6).

[15] Rizk H, Nishimur Y, Yamaguchi H, Higashino T. Drone-based water level detection in flood disasters. Int J Environ Res Public Health. 2021;19(1):237.

[16] Nanditha JS, Kushwaha AP, Singh R, et al. The Pakistan flood of August 2022: causes and implications. Earth’s Future. 2023;11(3):e2022EF003230.

[17] Ishtiaq N. The use of social media during the 2022 flooding in Pakistan: a study of the National Disaster Management Authority crisis communication. [Journal Name]. 2023.