KATHOLISCHE UNIVERSITÄT
EICHSTÄTT-INGOLSTADT

Masterarbeit

Wassererkennung auf Sentinel-1 Daten mithilfe eines Bayesian Convolutional Neural Networks

Anwendung von Unsicherheitsschätzungen zur Identifikation fehleranfälliger Bereiche und zur Verbesserung der Ergebnisse.

Water detection in Sentinel-1 data using a Bayesian Convolutional Neural Network

Application of uncertainty estimations to identify error prone areas and improve the results.

Mathematisch-Geographische Fakultät
Lehrstuhl für Physische Geographie
M. Sc. Umweltprozesse und Naturgefahren
6. Fachsemester

Autor:	Peter Mederer
Matrikel Nr.:	704311
Betreuer:	Dr. Tobias Heckmann
vorgelegt am:	31. 08. 2022

Abstract

Floods are a natural hazard that can seriously impact the affected communities. Therefore, improvements in flood management are necessary to better prevent and manage flood disasters. These can be achieved by mapping flooded areas using remote sensing data such as Synthetic Aperture Radar (SAR) data. SAR has the advantage of covering large spatial extents and operating weather and daylight independently. While conventional methods exist to detect water in SAR data, Convolutional Neural Networks (CNNs) have produced excellent results. The results, however, do not come without inaccuracies and uncertainties. Therefore, Bayesian Convolutional Neural Networks (BCNNs) have been developed to estimate the uncertainties of the model.

This study analyzes the conditions that prevail in misclassified areas. Certain landcover classes like bare soil show higher percentages of wrongly labeled pixels. The behavior of the estimated uncertainties is also tested over pixels that are wrongly and correctly labeled as well as over different landcover classes. It is found that uncertainties are higher over misclassified pixels and certain landcover types like bare soil and herbaceous vegetation. Based on the findings that uncertainties are elevated over falsely labeled pixels, the pixels are turned to their opposite class when exceeding an uncertainty threshold. After the relabeling, the performance metrics are compared to the initial metrics. In this study, multiple setups for relabeling are tested and compared. The approach is found to be working in certain areas.

The study is conducted to confirm the applicability of BCNNs to generate precise flood mapping products and to estimate model uncertainties. The relabeling also aims to shorten the process of training data creation. Training data creation is a resource-intensive step. By improving the results after the classification, less accurate training data might be usable to train the model. As a result, more training data can be efficiently generated to cover more expansive areas globally. The findings provide a basis to create more complete models in the future and further assist flood management.

Übersicht

Überflutungen stellen eine Naturgefahr dar, die weitreichende Folgen für die betroffenen Gemeinden haben kann. Zur besseren Prävention und zum besseren Verständnis der Naturgefahr werden Überflutungskarten genutzt. Diese können aus Fernerkundungsdaten, wie Synthetic Aperture Radar (SAR) Daten generiert werden. SAR hat den Vorteil, dass es große räumliche Flächen abdecken kann und wetter- und tageslichtunabhängig operiert. Es existiert eine Vielzahl herkömmlicher Methoden der Wassererkennung aus SAR Daten. In den letzten Jahren werden in der Literatur zunehmend Convolutional Neural Networks (CNNs) genutzt. Die Ergebnisse besitzen aber dennoch Ungenauigkeiten und Unsicherheiten. Um die Unsicherheiten eines Modells zu ermitteln, wurden Bayesian Convolutional Neural Networks (BCNNs) entwickelt.

Diese Arbeit analysiert die Gegebenheiten, die in fehleranfälligen Bereichen vorherrschen. Es wurde erkannt, dass einzelne Landnutzungsklassen, wie unbewachsener/leicht bewachsener Boden fehleranfälliger sind. Es wurde außerdem getestet wie sich die ermittelten Unsicherheiten über falsch und richtig klassifizierten Bereichen und über den einzelnen Landnutzungsklassen verhält. Es wurde herausgefunden, dass die Unsicherheiten über falsch klassifizierten Bereichen, sowie über Bereichen der Klassen von unbewachsenem Boden und Kräutervegetation, erhöht sind. Basierend auf der ersten der beiden Erkenntnisse, wurden Pixel oberhalb eines bestimmten Unsicherheitswertes in die gegenteilige Klasse umgewandelt. Danach wurden die Genauigkeitsmetriken mit den ursprünglichen Werten verglichen. Der Ansatz scheint für bestimmte Bereichen zu funktionieren. In dieser Studie wurden mehrere Modellsetups für den Versuch getestet.

Die Studie wurde durchgeführt, um die Genauigkeit der BCNNs zu prüfen. Das Umwandeln von Bereichen mit hoher Unsicherheit zielt auch darauf ab, den Prozess der Trainingsdatengenerierung zu verkürzen. Die Erzeugung von Trainingsdaten ist ein ressourcenintensiver Schritt. Durch die Verbesserung der Ergebnisse nach der Klassifizierung können weniger genaue Trainingsdaten zum Trainieren des Modells verwendet werden. Infolgedessen können mehr Trainingsdaten effizient erstellt werden, um größere Gebiete weltweit abzudecken. Die Ergebnisse bilden eine Grundlage für die Entwicklung umfassenderer Modelle in der Zukunft und zur Unterstützung des Hochwassermanagementes.

Acknowledgments

I want to thank all the people who supported me while creating this master thesis. Without you, the outcome would not have been possible.

I am incredibly grateful for the support from my primary supervisor at the German Aerospace Centre (DLR), Marc Wieland. Your knowledge and kind words helped me find a great topic and provided many insights along the way. I would also like to thank Candace Chow, who offered great input on various questions. I am also very grateful for the advice from Victor Hertel and Max Helleis. This work builds on the results of Victor's thesis, and he gave a great introduction to the topic, which helped me a lot. Additionally, I want to thank Sandro Martinis for providing the opportunity of creating the work at the DLR.

I also want to thank Tobias Heckmann, my supervisor at the university. He provided excellent advice about various questions and supported me well along the way.

Lastly, I want to thank my friends, family, and especially my girlfriend, who supported me on a daily basis.

Contents

Abstract II
Acknowledgments IV
Contents V
List of Figures VII
List of Tables IX
Abbreviations X
1 Introduction 11
2 Theoretical Background and the Current State of Research 13
2.1 Water Detection in Remote Sensing Data 13
2.1.1 SAR Functionality 14
2.1.2 Scattering Principles 15
2.1.3 Conventional Methods for water detection in SAR data 17
2.2 Advances through Deep Learning 18
2.2.1 Convolutional Neural Networks 18
2.2.1.1 Principle and Architecture 19
2.2.1.2 Convolutional Stage 19
2.2.1.3 Nonlinearity 21
2.2.1.4 Pooling Layer 21
2.2.1.5 Training 22
2.2.1.6 Water Detection in SAR Data 24
2.2.2 Advancements by Uncertainty Estimations 25
2.2.2.1 Aleatoric and Epistemic Uncertainty 26
2.2.2.2 Bayesian statistics and Bayesian inference 27
2.2.2.3 Bayesian Convolutional Neural Networks 28
2.2.2.4 Uncertainty estimations by BCNNs 30
2.2.2.5 Application of BCNNs and Uncertainty Estimations 32
3 Research Objectives 33
4 Data 34
4.1 Global Sentinel – 1 Dataset and Reference Data 34
4.2 Copernicus Global Landcover Data 36

5 Methodology 40
5.1 Setup of the Bayesian Convolutional Neural Network 40
5.2 Identification of error-prone regions of the BCNN 41
5.3 Uncertainty Analysis 42
5.3.1 Uncertainty Distribution over correctly and wrongly labeled pixels 43
5.3.2 Uncertainty Distribution over different types of landcover 44
5.4 Result optimization 44
5.4.1 Creation of label noise 45
5.4.2 Uncertainty estimation and Morphological filtering 46
5.4.3 Relabeling of the generated masks 47
5.5 Trained Models 49
5.6 Performance Metrics 50

6 Results 52
6.1 Initial Prediction Results 52
6.2 Detection of error-prone areas 54
6.3 Uncertainty Analysis 57
6.3.1 Uncertainty values over misclassified pixels 57
6.3.2 Uncertainty values over different landcover classes 67
6.4 Result optimization based on uncertainties 69

7 Discussion 75
7.1 BCNN performance 75
7.2 Error-prone areas of the BCNN 76
7.3 Uncertainty analysis 77
7.4 Applicability of relabeling based on uncertainties 79

8 Conclusions 81
8.1 Summary 81
8.2 Recommendations for Future Research 82

Bibliography 83
Appendix 95
A Test Scene Overview 95
B Accuracy and Certainty Maps 102

Declaration of Originality 143

List of Figures

Figure 1: Mechanisms of scattering over different land and water surface types.....15

Figure 2: A convolution executed with a 3 x 3 filter kernel and zero padding over a 5 x 5 input 20

Figure 3: ReLU function and the expected results based on the input being lower or greater than 0 21

Figure 4: Max pooling using a 2 x 2 filter kernel.....22

Figure 5: Deterministic sigmoid output 24

Figure 6: Aleatoric and Epistemic uncertainty.....26

Figure 7: Schematic structure of a simple ANN and BNN. 29

Figure 8: Deterministic sigmoid output and probabilistic sigmoid ensemble output 30

Figure 9: Uncertainty based on the spread of the sigmoid distribution 31

Figure 10: Uncertainty as the probability of a pixel being correctly labeled. 31

Figure 11: Global distribution of the training, validation, and test Sentinel-1 scenes 36

Figure 12: CNN U-Net architecture 40

Figure 13: Line simplification using the Visvalingam-Whyattt algorithm 46

Figure 14: Details of the relabeling step 48

Figure 15: Training and validation logs for the BCNNs trained with the whole dataset and the simplified data.....53

Figure 16: Initial prediction results for the trained BCNNs, trained and validated using the optimal data (Model A), the simplified data (Model B), and the simplified data with one epoch training (Model C). 54

Figure 17: Backscatter values over the 12 different landcover classes for all 18 test scenes. 56

Figure 18: Uncertainties over TP, FP, TN, and FN pixels for all 18 test scenes. 58

Figure 19: Overview of test scene 75 60
Figure 20: Overview of test scene 89 60
Figure 21: Visualization of highly uncertain areas for Scene 75 and UD1 61
Figure 22: Visualization of highly uncertain areas for Scene 75 and UD2 62
Figure 23: Visualization of highly uncertain areas for Scene 89 and UD1 63
Figure 24: Visualization of highly uncertain areas for Scene 89 and UD2 64
Figure 25: Visualization of highly uncertain areas for a zoomed-in detail in Scene 89 and for UD1 65
Figure 26: Visualization of highly uncertain areas for a zoomed-in detail in Scene 89 and for UD2 66
Figure 27: Distribution of the uncertainties for UD1 over the 12 landcover classes. 68
Figure 28: Performance metrics changes based on the chosen threshold for relabeling (0.01 steps) for Predictions 001, 003, and 005 70
Figure 29: Change in performance metrics compared to the initial performance after relabeling for all 18 scenes and Prediction 005 71
Figure 30: Relabeling for Scene 75 and 89 72
Figure 31: Relabeling for the zoomed-in detail of Scene 89 73
Figure 32: Maps of the conducted relabeling for scene 89 and a threshold of 0.05 74

List of Tables

Table 1: Factors leading to over- and underclassification of flooding areas in SAR data. The occurrence and impact of the respective factors are displayed 16

Table 2: Number of taken Accuracy Assessment samples, achieved overall accuracy and corresponding confidence intervals for each continent 38

Table 3: Summarized GLCS-LC100 landcover classes used in this study 39

Table 4: Hyperparameters used for the setup of the BCNN 41

Table 5: Quantitative metrics to evaluate classification performance 51

Table 6: Initial prediction results for the six predictions using the three models 53

Table 7: Overview of the proportion of pixels within each landcover class for the 18 test scenes 55

Table 8: Proportion of misclassified pixels over each landcover class to all misclassified pixels over all 18 test scenes and for all six predictions 56

Abbreviations

AC	Accurate and Certain
ANN	Artificial Neural Network
AU	Accurate and Uncertain
BCNN	Bayesian Convolutional Neural Network
BNN	Bayesian Neural Network
CNN	Convolutional Neural Network
ELBO	Log Evidence Lower Bound
FN	False-Negative
FP	False-Positive
GCLS-LC100	Copernicus Global Land Service Land Cover Map at 100m
GRD	Ground Range Detected
H	Horizontal
IC	Inaccurate and Certain
IoU	Intersection-over-Union
IQR	Interquartile Range
IU	Inaccurate and Uncertain
IW	Interferometric Wide Swath
KL	Kullback-Leibler
NDWI	Normalized Difference Water Index
NRCS	Normalized Radar Cross Section
P	Probability
Q1	Quartile 1
Q3	Quartile 3
ReLU	Rectified Linear Unit
SAR	Synthetic Aperture Radar
TN	True-Negative
TP	True-Positive
UAV	Unmanned-Arial-Vehicle
UD1	Uncertainty Definition 1
UD2	Uncertainty Definition 2
V	Vertical

1 Introduction

Natural hazards can have an immense impact on human lives and activities. They are associated with natural processes that can have extensive consequences for humans, properties, and economies. Affected communities often cannot compensate for the damages and rely on outside help. Natural hazards can appear in different forms, like earthquakes, tsunamis, volcanic eruptions, or flooding (Bokwa, 2013; Bryant, 2005). Floods pose one of the most impactful natural hazards on a global scale. They are defined as an increase in water level and thus inundating areas usually water-free (Merz et al., 2010; Tsakiris, 2014). Damages to communities occur when there is an interaction between the flood and socio-economic structures. The level of damage is determined by the number of affected humans and the property value (Barredo, 2009; Mederer, 2020). According to the Emergency Events Database, riverine flooding events caused financial damages of almost 461 billion US-Dollars and killed nearly 70 thousand people worldwide in the years 2000 to 2022 (EM-DAT, 2022). The impact of climate change on future global flooding activity has been widely researched. An intensification of the hydrological water cycle is expected for all global climate regions. The higher temperatures provide a higher water availability over large parts of the globe. Therefore, an increase in the frequency and intensity of heavy rain events and resulting floodings is expected during the following centuries (Tabari, 2020).

Flood assessments can help avoid possible damage and decrease the impact of hazardous flooding events. In addition, they provide a data basis for first responders and for other forms of disaster management. For example, the size of the flooded areas can be extracted to assess a flood. This provides information about the event's location, extent, and spatial distribution (Mudashiru et al., 2021). There are several possible applications of flood mapping products, one being the near real-time generation of flooding maps. They can warn residents and assist first responders (Z. Li et al., 2017; Martinis et al., 2009). Another possibility is to create maps after the flood to analyze the event's impact and consequences (van der Sande et al., 2003).

Remote Sensing data have been widely used as a basis for flood assessment, due to the extensive spatial coverage of the data, especially in satellite remote sensing imagery. The sensors receive electromagnetic waves and allow assumptions about the imaged surface.

Two operational types of remote sensing are used for water surface mapping. Optical systems are passive as they only receive radiation in the visible and infrared range (K. Li et al., 2020). Radar systems, especially Synthetic Aperture Radar (SAR), actively emit and receive radiation in the microwave range. SAR is widely used for flood mapping as it operates weather- and daylight-independent. As water possesses a smooth surface, the radiation is reflected away from the sensor resulting in low backscatter values returning to the sensor, thus appearing dark in the image. This is the basic assumption of water extraction from SAR data (Martinis, Kuenzer, & Twele, 2015). While various conventional methods exist, Deep Learning approaches have gained popularity in recent years. Convolutional Neural Networks (CNNs) seem well suited for water mappings they speed up the mapping process (Helleis et al., 2022). While the developed CNNs achieve high overall accuracies, some areas are error-prone. This might be due to similar backscatter properties of smooth surfaces leading to overestimations. Other factors, like submerged vegetation, have been found to lead to underclassifications as the detectable water surface appears smaller (Martinis, Kuenzer, & Twele, 2015). Conventional CNNs also do not provide any information about the model's uncertainty per pixel. Their only result is a binary water mask. Bayesian Convolutional Networks (BCNNs) have been developed to account for this problem and offer data about the uncertainties. They utilize Bayesian statistics to turn the deterministic CNN output probabilistic. The uncertainties can be calculated based on the probabilistic output (Hertel, 2022). However, little research has been conducted about the properties and the applicability of the generated uncertainties.

This study examines the spatial distribution and preexisting conditions of error prone areas and uncertainties of a BCNN built to detect water surfaces in SAR data. It further investigates how uncertainties can be used to improve the classification results. This presumes that uncertainties are higher over misclassified pixels. The following research aims to confirm this assumption. By improving the results based on uncertainties, less accurate training data might be used. This could lead to lesser resources needed to generate adequate reference data.

2 Theoretical Background and the Current State of Research

The following chapter highlights the theoretical basis of Synthetic Aperture Radar (SAR) and different image segmentation techniques used to identify water surface areas. This includes a review of relevant studies. The chapter focuses on deep learning approaches used for image segmentation; the findings serve as a basis for the experiments conducted in this thesis.

2.1 Water Detection in Remote Sensing Data

Remote Sensing data have been widely used in image segmentation for water area detection (Brivio et al., 2002; Klemas, 2015; Sanyal & Lu, 2004a). Remote sensing sensors are located at a distance to the study object. They are usually satellite-, aircraft-, or Unmanned Aerial Vehicle (UAV) - borne. Backscattered electromagnetic radiation detected by these sensors can be observed. There is a distinction between optical and radar systems, which are used to detect flooded areas and thus provide the data foundation for flood mapping (Anusha & Bharathi, 2020; Markert et al., 2018; Mederer, 2020). However, it should be noted that remote sensing data can only be used to derive the extent of the water surface, not the actual extent of flooding. Therefore, to detect flooded areas, the data must be compared with reference data showing the water body at normal water level (Martinis, Kuenzer, Wendleder, et al., 2015; Mederer, 2020).

Optical systems can detect wavelength ranges between visible and infrared light. It is a passive remote sensing system, meaning the sensor emits no radiation. Optical sensors divide these wavelengths into channels that delineate the wavelength ranges from one another (K. Li et al., 2020). Optical images can be used to visually interpret and manually map the boundaries between flooded and water-free areas. Infrared channels are particularly suitable for this purpose (Deutsch & Ruggles, 1974; Mederer, 2020; Moore & North, 1974). There also exists a variety of spatial indices used in optical water detection. Here different bands are offset against each other (e.g., Normalized Difference Water Index) (Jain et al., 2005; Mederer, 2020; Moore & North, 1974; Munasinghe et al., 2018; Suwarsono et al., 2013).

Additionally, SAR data has been used for water surface detection. SAR systems provide the substantial advantage of operating weather and daylight independently. Flooding often occurs due to heavy rainfall accompanied by high cloud cover. In contrast to optical sensors, SAR waves are capable of cloud cover penetration. As a result, almost all obtained SAR data can be used for flood detection and is, therefore, often used for rapid flood mapping applications (Lakshmi, 2017; Martinis, Kuenzer, Wendleder, et al., 2015; Mederer, 2020).

2.1.1 SAR Functionality

SAR sensors emit and receive electromagnetic radiation pulses in the microwave range (1 mm - 1 m wavelength). They operate in different frequency ranges of the radar spectrum, which are divided into so-called bands. Sentinel-1, used in this study, operates in the C-band, corresponding to a wavelength of approximately 5.6 centimeters. In contrast to passive systems, it is an active radar system and can send radar beams to specific areas. This allows a high temporal and spatial resolution of investigation areas (ESA, 2021; Mederer, 2020). The detected radiation contains information about intensity and phase difference. In this study, only information about intensity is used to derive water masks. SAR systems artificially increase their spatial resolution by summing the echoes of several acquired radar echoes, thus simulating an extension of the actual antenna. The result is an increase in the spatial resolution in the direction of flight (Bamler, 2000).

The applicability for water detection is also influenced by other technical properties like the chosen frequency band or the polarization. SAR can emit and receive radiation in Horizontal (H) and Vertical (V) polarization. This leads to possible emit-receive combinations of HH, VV and the cross combinations HV and VH (Hertel, 2022; Manavalan & Ramanuja, 2018; Mederer, 2020). Based on the object of study different polarizations might be better suited. In this study, VV and VH polarized data are processed and analyzed to recognize a broader spectrum of properties. The combination has been found to contain optimal information for water detection (Helleis et al., 2022). The Sentinel-1 data used in this thesis was acquired using the Interferometric Wide Swath (IW) mode. Here the data is acquired by a 250 km swath (ESA, 2021; Hertel, 2022).

2.1.2 Scattering Principles

Depending on the type and structure of the surface, altered signals arrive at the sensor. This is because different reflection and scattering mechanisms occur. For example, specular reflections appear over open, smooth water surfaces, reflecting the radiation away from the sensor. As a result, radiation of lower intensity returns to the sensor. As a result, the water surface appears in dark coloration and high contrast compared to the surrounding areas (Manavalan & Ramanuja, 2018). Based on this fact, water is detected in SAR data. Figure 1 illustrates the effects of different forms of backscattering as well as the interaction with different land surface types. In addition, it is visible that diffuse scatter mechanisms (Diffuse surface scattering, diffuse volume scattering, brag scattering) over land as well as over water alter the returned backscatter. These may lead to over- and under-classifications of the water extent.

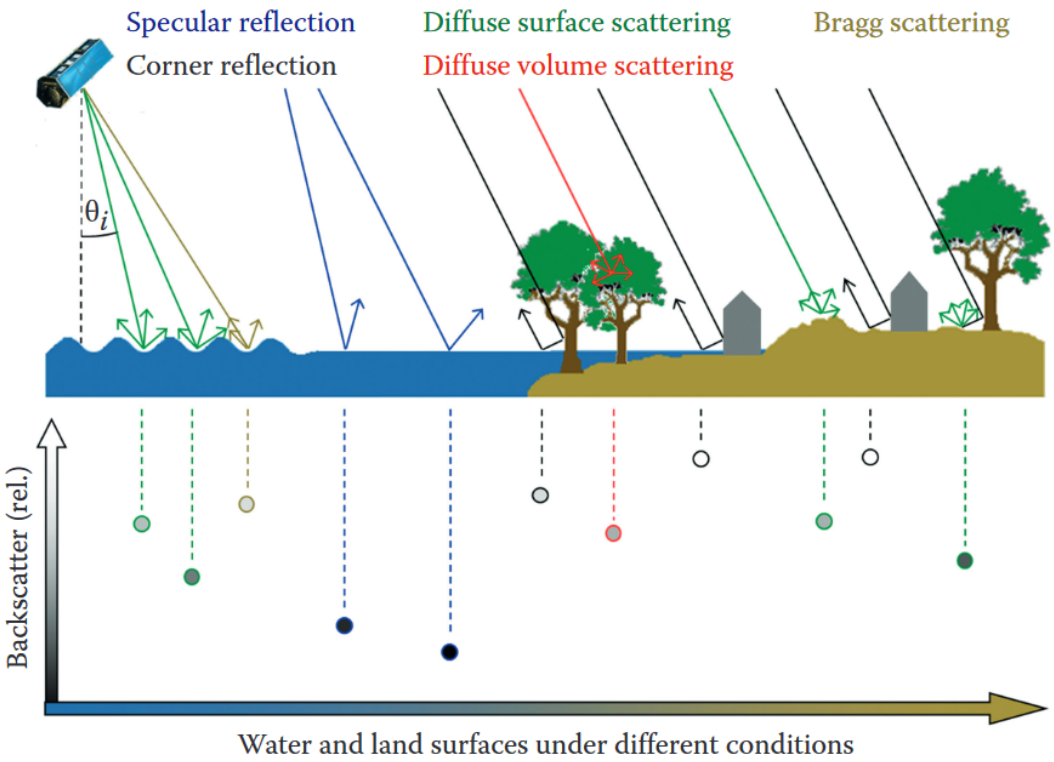


Figure 1: Mechanisms of scattering over different land and water surface types. Diffuse and specular components of surface-scattered radiation as a function of SAR incidence angle and surface roughness (Martinis, Kuenzer, & Twele 2015).

Table 1 displays the factors leading to misclassifications of water extents in SAR data and their occurrence and impact on the flood classification result. Overclassifications with

the highest impact lie over smooth, natural surfaces (e.g., sand dunes, bare ground). The areas are confused for water due to their similar backscatter characteristics. The other central erroneous region appears behind tall structures like mountains or buildings. This issue intensifies the higher the recording angle is. No signal reaches the surface resulting in dark areas on the image. Underestimations happen mainly in flooded areas, partly or wholly covered by vegetation. The signal never reaches the water surface (volume scattering at tree crowns) or is reflected to the sensor by corner reflections (Martinis, Kuenzer, & Twele, 2015).

Table 1: Factors leading to over- and underclassification of flooding areas in SAR data. The occurrence and impact of the respective factors are displayed (changed following Martinis, Kuenzer, & Twele (2015).

Overestimation of Flooding	
Factor	Occurrence / Impact
Shadowing effects behind vertical objects (e.g., vegetation, topography, anthropogenic structures)	+++
Smooth natural surface features (e.g., sand dunes, salt and clay pans, bare ground)	+++
Smooth anthropogenic features (e.g., streets, airstrips)	++
Heavy rain cells	+

Underestimation of Flooding	
Factor	Occurrence / Impact
Volume scattering of partially submerged vegetation and water surfaces completely covered by vegetation	+++
Double-bounce scattering of partially submerged vegetation	++
Anthropogenic features on the water surface (e.g., ships, debris)	+
Roughening of the water surface by wind, heavy rain, or high flow velocity	+
Layover effects on vertical objects (e.g., topography urban structures, vegetation)	+

Note: Feature range: high +++; medium ++; low +

Backscatter effects also vary based on the SAR wavelength the sensor is operating in. Longer wavelengths can penetrate canopy covers. This results in possible scattering effects with branches. As they are penetrating the canopy cover to a lesser degree, an increased effect of volume scattering is detectable for shorter wavelengths. Built-up areas are also prone to underclassification. Reasons are the strong double-bounce effects that occur whether the area is flooded or not (Martinis et al., 2009; Martinis, Kuenzer, &

Twele, 2015). Figure 1 also illustrates the relative backscatter intensity returning to the SAR sensor based on surface and backscatter properties. Again, the lower intensity of open water areas emerges.

2.1.3 Conventional Methods for water detection in SAR data

Therefore, the lower backscatter intensity over open water is utilized to map water extents. The usage of SAR data for flood detection started with visual interpretation and manual digitization of the floodplain (Mederer, 2020; Sanyal & Lu, 2004). Disadvantages are the high time consumption, subjectivity of the creator, and poor reproducibility of the results. These disadvantages also limit the use of this technique for rapid flood mapping as it is barely viable for the rapid digitization of larger areas (Manavalan, 2017).

Another digital method to map water extents in SAR data is pixel thresholding (Martinis, Kuenzer, & Twele, 2015). Most water classification methods perform a binary classification based on the backscatter value of each pixel. Each pixel of the SAR data set has a value of the measured amplitude or intensity. A threshold value is chosen to separate the pixels that represent water and those that do not. All pixels below that value are classified as water. The quality of the threshold-based classification depends on the contrast between water-covered and water-uncovered areas. Due to waves, the increased surface roughness of water can lead to poorer classification results (Manavalan & Ramanuja, 2018; Martinis, Kuenzer, & Twele, 2015). One possibility to determine the threshold is supervised determination. Here, the global histogram of the image is manually checked for the appropriate threshold value using a ‘trial and error’ approach. Then, the threshold is adjusted until a satisfactory classification result is obtained (Brivio et al., 2002; Martinis, Kuenzer, Wendleder, et al., 2015; Mederer, 2020). An advancement of this approach is the empirical determination of the threshold value. Here, the threshold is calculated based on the statistics of the SAR (Gstaiger et al., 2012; Kuenzer et al., 2013). Furthermore, approaches to calculate the threshold automatically can be found in the literature. These can be particularly useful for the rapid generation of inundation maps in the event of a crisis (Martinis et al., 2009; Matgen et al., 2011; Mederer, 2020).

Another approach is the Change Detection method. This method compares multitemporal images of the floodplain with reference datasets generated before the flooding event. A possible approach here is amplitude change detection. Areas are determined as flooded if

a significant decrease in intensity is discovered (Manavalan, 2017; Sumaiya & Shantha Selva Kumari, 2018).

Object-based classification is another technique that is being used. However, only a limited number of studies have been conducted for water detection in SAR data. In object-based classifications, individual pixels do not contain information necessary for the classification. Instead, the SAR data is divided into segments of multiple pixels containing homogenous information (Heremans et al., 2003; Herrera-Cruz & Koudogbo, 2009; Mederer, 2020).

2.2 Advances through Deep Learning

While there exists a variety of different approaches for water detection with SAR data, an increase in studies applying Deep Learning methods, especially CNNs, has been observed. The following sub-chapter addresses the theoretical background behind the utility of CNNs for water extent detection. In particular, the benefits of using a BCNN are highlighted. The findings from the review of these studies are used for further analysis in this thesis.

2.2.1 Convolutional Neural Networks

CNNs are a form of Artificial Neural Networks (ANN) and belong to the domain of Deep Learning. Deep Learning is one of the latest advances in Machine Learning and is a form of Artificial Intelligence (AI). Machine learning approaches aim to identify patterns and regularities by employing existing data and algorithms. The generated knowledge can be generalized based on the data used to train the model. The models can then be used for problem-solving and analyzing previously unknown data (Gevrey et al., 2003). The difference between Deep Learning and conventional Machine Learning approaches is how the models learn. Deep Learning methods try to simulate parts of the human brain's functionality to process information. This works via connections of artificial neurons. Depending on the data used for training, the Neural Network can adapt and create new connections between neurons. Deep Learning uses many layers to learn features of the data that can be adapted to the data by repeated training. Traditional Machine Learning methods use handcrafted features (e.g., vegetation indices, texture). Deep Learning and

Machine Learning methods can be trained in an unsupervised, supervised, or semi-supervised manner. Deep Learning models are especially suitable for large amounts of data (Abiodun et al., 2018; Gevrey et al., 2003; Wang, 2003).

2.2.1.1 Principle and Architecture

CNNs are a particular form of ANNs and are well suited for object recognition in images. The networks utilize the mathematical operation called convolution. The data supplied to the model passes a series of layers. A convolutional layer consists of a convolutional stage, applied non-linearity, and a pooling stage. During training, the CNN learns the values of the filter kernel used for the convolution via backpropagation. The architecture of a CNN is defined by the sequence of different convolutional layers the data passes. Further information about different CNN architectures used for water detection in SAR data can be found in Helleis et al. (2022). The following sections provide an overview of the structure of a convolutional layer (Helleis et al., 2022; Hertel, 2022). They also aims to explain the functionality of a CNN and the way it behaves during training and prediction. The available data is split into training, validation, and test data. The training data is used for the actual training, and the validation data test the improvements in the performance of the model during the training process. The test data is used to evaluate the model performance after the training, containing data unknown to the model (Keiron & Nash, 2015). All datasets consist of the input and reference data to which the output is compared.

2.2.1.2 Convolutional Stage

During the convolution, a filter is applied pixel-wise over the input image. The filter is a kernel consisting of a two-dimensional array of 0 and 1. The filters and their values are also referred to as the model's weights (Albawi et al., 2017; Keiron & Nash, 2015). An element-wise product between the filter kernel and the pixels lying beneath is calculated for each pixel. The resulting values are summed up, and the filter is moved to the next pixel. Based on the filter kernel structure, different image properties might be extracted. For example, one filter can be applied for edge detection while another extracts other texture details (Albawi et al., 2018; Gu et al., 2018). Thus, the application of the filter integrates not only information about the pixel itself but also about the surrounding pixels. The resulting weighted summation is displayed in a feature map. The convolution is specified by stride, kernel size, and zero padding. Stride determines the step size at which the filter is moved. For example, a value of stride = 1 means the filter is moved one pixel at

a time. The filter size can strongly impact the outcome and must be fixed across all operations in one convolution. Zero padding defines the number of rows and columns containing zeros to be added around the input, thus determining the size of the resulting feature map. Without zero padding, the feature map size would shrink after each convolution, limiting the number of convolutions in a network (Albawi et al., 2017; Helleis et al., 2022; Keiron & Nash, 2015). Figure 2 illustrates the element-wise multiplication and summation of the input data, the filter kernel, and the resulting feature map.

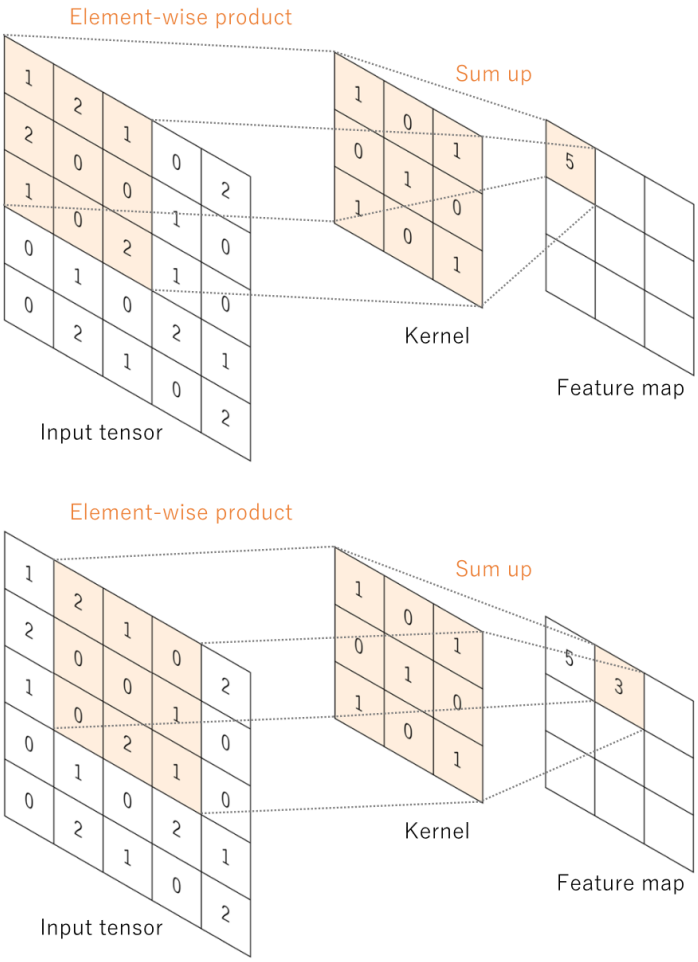


Figure 2: A convolution executed with a 3 x 3 filter kernel and zero padding over a 5 x 5 input (Yamashita et al., 2018).

2.2.1.3 Nonlinearity

In the next step, a nonlinearity function, a so-called activation function, is applied to the convolution output. The reason to add nonlinearity is to adjust or cut off the output. It is a vital part of the neural network. The most used function is the Rectified Linear Unit (ReLU) function which is defined as:

$$a_{i,j} = max\ (z_{i,j}, 0)$$

$z_{i,j}$ being the input of the ReLU function at location (i,j). The ReLU function turns the negative part to zero and retains the positive part of the input (Albawi et al., 2017; Gu et al., 2018; Hertel, 2022). Figure 3 illustrates the ReLU function and the expected output based on the input being lower or greater than 0.

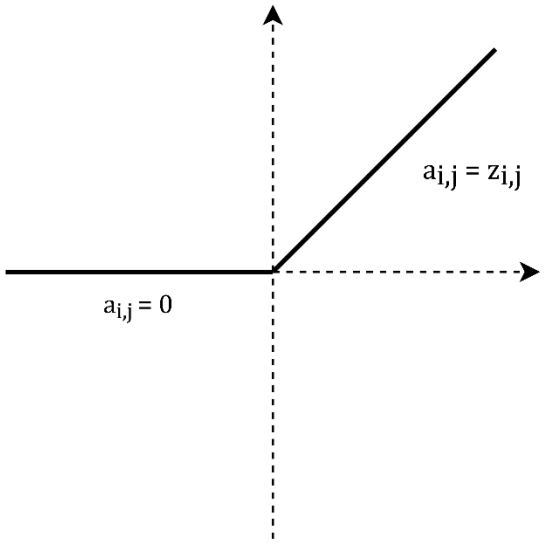


Figure 3: ReLU function and the expected results based on the input being lower or greater than 0 (created according to Gu et al. 2018).

2.2.1.4 Pooling Layer

Subsequently, the pooling layer reduces the dimension of the output while retaining the features in the feature map. This also decreases the disk space and computing power needed for further processing. A commonly used approach is max pooling, as illustrated in Figure 4. The highest value inside the 2 x 2 pooling filter proceeds to the output layer. Various pooling methods exist, such as max pooling drop-out, averaging, or summation. However, in the literature, max pooling is most commonly used (Helleis et al., 2022; Yamashita et al., 2018).

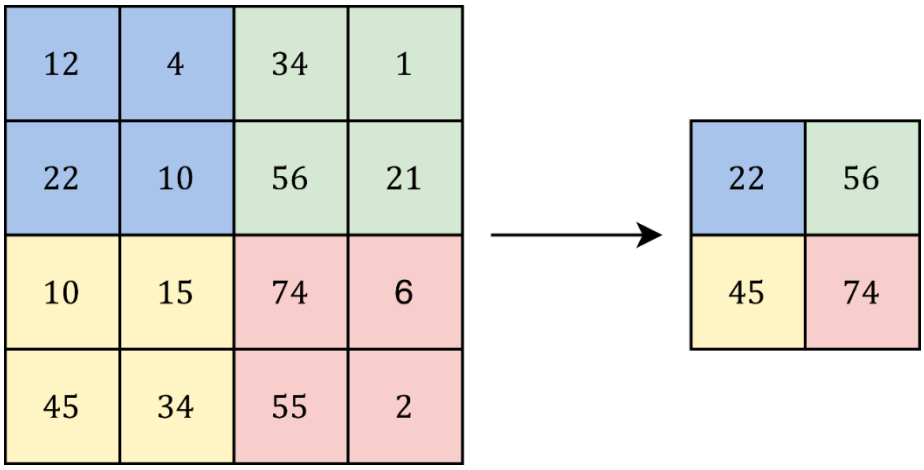


Figure 4: Max pooling using a 2 x 2 filter kernel.

2.2.1.5 Training

Consecutive convolutional layers construct the CNN. The convolutional layers consist mainly of the convolutional stage, the activation function, and the pooling stage. The filter kernels are adjusted during training to generate a result that matches the training data. The method that is often deployed to update the filters is backpropagation. Here the filter values are adjusted to minimize the difference between the output of the model prediction and the desired output. This difference is measured using a loss function, also called the training loss. The training loss is a metric to assess the model's goodness of fit to the training data. It is calculated by the sum of errors for every batch in the training data. The definition of a batch is explained below. The direction in which the filter values need to be adjusted is determined by the gradient of the loss function (Helleis et al., 2022; Hertel, 2022; Rumelhart et al., 1986). Cross-entropy and focal loss functions are often used in the literature. Cross-entropy loss is conventionally used in classification by deep learning models. However, the function tends to perform poorly when there is an imbalance in class distribution. This presents a problem for flood detection in SAR data as the water class is frequently underrepresented.

To overcome this problem, weighted cross-entropy functions were developed. Other loss functions like weighted cross-entropy loss or focal loss also aim to minimize the problem (Hertel, 2022; Yamashita et al., 2018). Additional to the training loss, the validation loss is calculated and shows a similar metric. The difference is that the validation loss is calculated for the validation data that was not part of the training process after each epoch. Epochs are a hyperparameter that is further explained in the following subsection. The

validation loss shows the model's improvement after the weights were adjusted based on the training loss. Furthermore, the training and validation accuracies can be extracted. They show how well the model predicts the training data (it is trained with) as well as the validation data (it has not seen before) between each epoch (Albawi et al., 2017; Keiron & Nash, 2015).

Hyperparameters are used in the setup of a CNN that greatly influence the desired output. One is the learning rate (i) controlling the rate at which the weight values should be changed every time they are updated (Jacobs, 1988). The learning rate decay (ii) determines how much the learning rate should decrease after a certain amount of epochs training (You et al., 2019). The number of epochs (iii) defines the number of repetitions for which the whole training data set is trained. Since updating the weights (filter values) is an iterative process, more than one epoch training might be necessary. Choosing the correct number of epochs also helps in reducing underfitting and overfitting. Both lower the model's ability to generalize from the training data to data the model has not seen before. Overfitting means the model is too well adapted to the training data, so data outside that dataset cannot be predicted well. This happens when the training is conducted for too many epochs. This is also demonstrated by a divergence of training and validation loss. Overfitting is visible when the training loss decreases further while the validation loss increases again. For too few epochs, the opposite occurs, defined as underfitting. The model is not well enough adapted to the training data. As a result, the model cannot generalize to data outside the training data (van der Aalst et al., 2010). To avoid overfitting and decrease the probability of underfitting, an early stopping (iv) parameter is set. When there is no increase in performance metrics after a set number of epochs, the model finishes the training process. This is measured by comparison of the validation losses. Another parameter to be set is the batch size (v). It determines the number of tiles used for training at a time. After one batch, the next one is looked at. The literature generally agrees that the optimal batch size for CNNs is between 64 and 512. The value is usually set to a power of 2 (He et al., 2016; Simonyan & Zisserman, 2014). This is explained by optimized matrix libraries working most efficiently if a power of two is chosen (Martin et al., 2013). Some studies use a batch size of a multiple of 10. Radiuk (2017) suggests that further investigations on the optimal batch size should be conducted. The batch size can also be limited by the physical capabilities of the used computer, as larger batch sizes require higher amounts of Rapid Access Memory (RAM) in the Graphics Processing Unit

(GPU) or the Central Processing Unit (CPU) used for training and predictions of the CNN (Mustafa et al., 2019). Further information regarding the functionality of CNNs can be found in Albawi et al. (2017).

2.2.1.6 Water Detection in SAR Data

The output generated by a CNN is usually a deterministic sigmoid output. This is the case since a sigmoid function is often used as the activation function in the last convolutional layer. Consequently, the model's prediction generates a single sigmoid output for every pixel. Figure 4 illustrates the deterministic sigmoid output. The Sigmoid values stored in each raster output cell should not be taken as a direct representation of probabilistic uncertainty or as frequentist probabilities. Instead, the values describe how closely a given pixel classification matches the training distribution. Values closer to 1 are nearer to the class water. For values closer to 0, it is the opposite case. Values around 0.5 are assumed to be unclear pixels. Generally, a threshold of 0.5 is chosen, and pixels exceeding this threshold are labeled as class water (Helleis et al., 2022; Hertel, 2022).

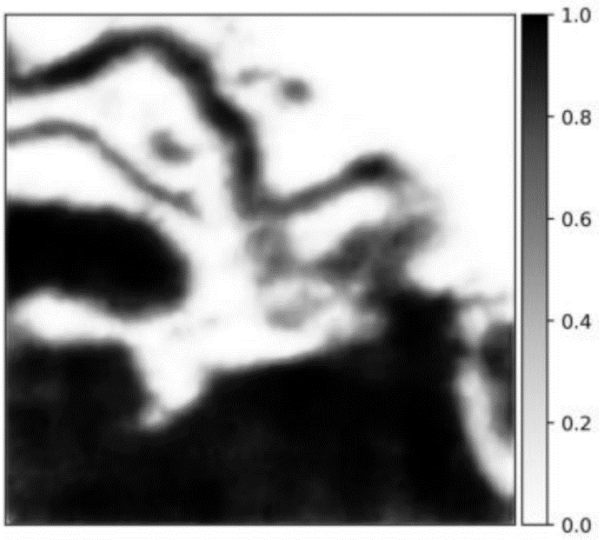


Figure 5: Deterministic sigmoid output (Hertel, 2022)

The functionality of this approach has been tested very successfully in multiple scientific studies (B. Liu et al., 2019; Nemni et al., 2020). CNNs experienced a substantial increase in usage to answer various geophysical research questions. There appears to be good applicability to detecting water surfaces in SAR data using CNNs. The methodology has outperformed various conventional methods (Helleis et al., 2022). Using CNN, Liu et al.

(2017) was the first study to generate flooding masks from Radarsat and ERS-1 data directly. According to Helleis et al. (2022), Kang et al. (2018) described the importance of using different scenes as the training, validation, and test data to avoid spatial autocorrelation. Liu et al. (2019) tested the impact of using different polarizations as inputs. They concluded dual polarizations and VH polarization to produce the most accurate results. Nemni et al. (2020) tested the approach, compared to most prior studies, over different geographic conditions globally. They also tested multiple CNN architectures and concluded that the U-Net architecture is most suitable for water surface and flood detection in SAR data. This hypothesis has been confirmed by multiple other studies (Bonafilia et al., 2020; Muñoz et al., 2021; Pai et al., 2020). Most recently, Helleis et al. (2022) compared the effectiveness of multiple CNN architectures for water surface detection and compared the results to an operational rule-based processor. The rule-based processor conducts water mapping by automatically thresholding the SAR data (Twele et al., 2016). The CNN models outperformed the rule-based processor in all conducted experiments. However, water surface detection in SAR data faces multiple challenges. These occur with the use of conventional methods as well as with CNNs. Multiple conditions hinder correct classifications in certain areas. These errors have been recognized to originate from different geographical and physical sources. Water detection relies on a sufficiently high contrast between water and non-water areas. Water is commonly detected via thresholding as the water generally corresponds to low backscatter values (as mentioned in section 2.1.2). However, this is also a characteristic of sand areas in arid regions, which may lead to incorrectly detected water pixels (Martinis et al., 2018). Other challenging land covers and uses have also been identified, including mountainous regions, due to the presence of radar shadows, built-up areas, and submerged vegetation, among others (Bertram et al., 2016; Helleis et al., 2022; O'Grady et al., 2011; Westerhoff et al., 2013). While those regions are often mentioned in studies, few attempts have been made to quantify these areas. Expanding knowledge about preexisting conditions of error-prone areas is vital to further improve classification results in those areas. Additional details about the advancements of CNNs for water surface detection can be found in Helleis et al. (2022).

2.2.2 Advancements by Uncertainty Estimations

CNNs produce a deterministic sigmoid output. This does not consider any uncertainties in the prediction, as only binary watermasks are created with a sharp border between

water and non-water classes. Uncertainties may originate from different sources. The quality of the training data strongly depends on their way of creation. Errors in humanmade and automatically generated watermasks can occur as both rely on the expertise and accuracy of their producers. The addition of uncertainties to the binary watermask might prove reasonable. Falsely labeled areas could have severe consequences when provided as a near real-time flooding product. BCNNs might provide a method to generate uncertainty estimations for deep learning approaches (Hertel, 2022).

2.2.2.1 Aleatoric and Epistemic Uncertainty

Different types of uncertainties mentioned in the literature should be elaborated to better understand the uncertainty that can occur in a CNN. In regression modeling, two types of uncertainty arise called aleatoric and epistemic uncertainty. Aleatoric uncertainty is assumed to be the randomness of the data that the model cannot explain. Additional data and information are not able to reduce this uncertainty. Contrarily, a lack of knowledge or data is presumed to be the reason for epistemic uncertainties. The addition of further information might be able to decrease this uncertainty. For most applications in Machine Learning, only a limited amount of data is available. Thus, epistemic uncertainty can only be decreased to a certain point (Hertel, 2022; Hüllermeier & Waegeman, 2021; Kiureghian & Ditlevsen, 2009). Figure 6 provides a schematic overview of both types of uncertainty. Even though sufficient information is available in the left figure, the prediction at the marked point possesses an aleatoric uncertainty. This is due to the overlapping of the two classes. In the right figure, a lack of available data causes a lack of knowledge about the correct hypothesis. This is expressed as a case of epistemic uncertainty (Hüllermeier & Waegeman, 2021).

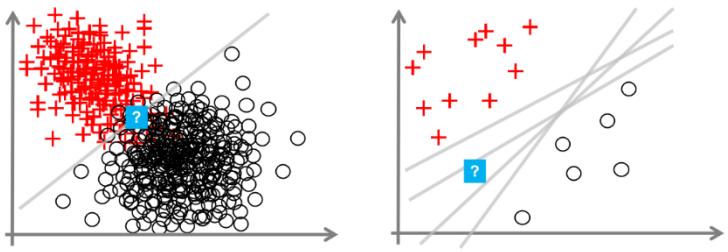


Figure 6: Aleatoric and Epistemic uncertainty: Even though enough information exists about the optimal hypothesis, the prediction at the marked point possesses an aleatoric uncertainty as the classes overlap (left); A lack of available data causes a lack of knowledge about the optimal hypothesis, defined as epistemic uncertainty (Hüllermeier & Waegeman, 2021).

A way to capture the aleatoric uncertainty is by calculating the range of +/- 1 or 2 standard deviations from the mean. This assumes that a relationship between two variables exists. The epistemic uncertainty can be estimated by expressing the weights of a model as a posterior probability distribution curve. The weights are randomly picked based on the distribution curve, and the output varies for each prediction. As a result, the model becomes probabilistic. Modern machine learning frameworks allow a combination of those two uncertainties. This approach can then be used for uncertainty estimation in CNNs (Hertel, 2022; Sountov et al., 2019).

2.2.2.2 Bayesian statistics and Bayesian inference

The uncertainty estimations generated for CNNs are based on Bayesian statistics. Bayesian statistics are based on the Bayes' theorem and are used in data analysis to update the available knowledge of a parameter with the information provided by observed data. This available knowledge is captured by the so-called a priori distribution. A parameter's actual value is considered random since the value is uncertain. Therefore, the prior distribution and the information added by new data are used to form a posterior distribution. This opens the possibility of directly using the rules of probability to make inferences about the parameter (Bolstad & Curran, 2016; van de Schoot et al., 2021). The Bayes' Theorem was first mentioned in an essay by Thomas Bayes in 1763 (Bayes, 1763).

The Theorem relates conditional probabilities. When given two events A and B, $P(A|B)$ describes the conditional probability of B happening, given that A happens. Bayes' theorem creates a relation between the two conditional probabilities $P(A|B)$ and $P(B|A)$. In the utilization for Neuronal Networks, the posterior distribution describes the conditional probability between a model parameter θ and the provided data γ (Bolstad & Curran, 2016; Hertel, 2022; van de Schoot et al., 2021). Bayes' Theorem is, in this case, described as:

$$P(\theta|\gamma) = \frac{P(\gamma|\theta)P(\theta)}{P(\gamma)}$$

Where

$P(\theta)$ is the prior distribution; it does not include any information about the observed data γ.

$P(\gamma)$ is the prior distribution of the observed data γ and acts as a normalizing constant.

$P(\theta|\gamma)$ is the conditional probability of θ, given γ. This is the posterior distribution as the model parameter θ is calculated based on the provided data γ.

$P(\gamma|\theta)$ is the conditional probability of γ given θ. This represents the likelihood function.

Bayesian inference describes the probability of the hypothesis being true based on the added data. In the case of the deep learning model, the target is to compute the true posterior distribution of a parameter based on the added data. The function of calculating the posterior distribution is intractable, as Shridhar et al. (2019) mentioned. Thus, an approximation of the true distribution is necessary. This can be achieved by using variational inference. It is possible to approximate the distribution by utilizing a finite number of variables. As the approximate distribution needs to be as close as possible to the true one, the Kullback-Leibler (KL) divergence is introduced. KL divergence is used to identify the resemblance of two distributions and is tried to be reduced as much as possible. However, it still contains intractable elements. To resolve this problem further, it has been proven that minimizing the KL divergence corresponds to maximizing the log evidence lower bound (ELBO) (Hertel, 2022; Shridhar et al., 2019). The variational inference may be combined with backpropagation as the Bayes by Backprop algorithm introduced by Blundell et al. (2015). This means the filter parameters may be represented as a prior distribution instead of fixed filter values and can be updated using the Bayes variational inference to approximate the posterior distribution.

2.2.2.3 Bayesian Convolutional Neural Networks

BCNNs are based on a probabilistic approach towards ANNs and, more specifically, CNNs. ANNs are trained deep learning models, and during the training process, the weight of each neuronal connection is learned to generate the desired output. This commonly happens using the backpropagation algorithm (Gu et al., 2018; Jospin et al., 2022; Simonyan & Zisserman, 2014). The weights are updated until a model matching all specifications set by the hyperparameters is created. The weight values are fixed values after training. Given the same input for the prediction, the trained model will always generate an identical output (Keiron & Nash, 2015). Bayesian Neural Networks (BNNs) integrate a stochastic element. By utilizing Bayesian statistics, the model is not learning a fixed

weight value but rather the weight over a probability distribution function. This function acts as the prior distribution. During the training process, this distribution is updated using Bayesian inference to obtain a posterior probability distribution. As the function is intractable, a variational inference approach is conducted to approximate the true posterior distribution (Hertel, 2022; Jospin et al., 2022). Once updated, the generated posterior distribution acts as the prior distribution for the next training step. Figure 7 shows the schematic difference between the neurons of a conventional ANN and a BNN. Here the weights are not one fixed value, but the weight values get randomly picked in accordance with the probability distribution. This generates altering outputs for every prediction of the same input turning the model probabilistic (Gal & Ghahramani, 2015; Jospin et al., 2022).

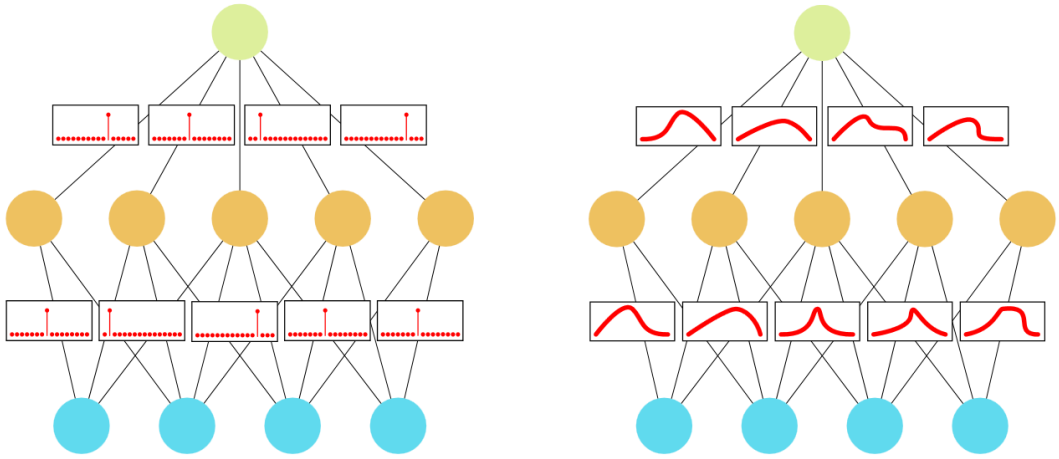


Figure 7: Schematic structure of a simple ANN and BNN. The weights are fixed values for the ANN, and for the BNN, values are along a probability distribution (Jospin et al., 2022).

For CNNs, the weights are not single values but filter kernels that perform the mathematical operation of convolution. Those filter kernels are updated during training. BCNNs fuse the concepts of CNN and BNN. Therefore, the kernel values are determined as distribution curves. These get updated during training via the Bayes by Backprop algorithm. In this case, the model also possesses a slightly different model setup for each prediction resulting in differing results (Blundell et al., 2015; Gal & Ghahramani, 2015; Hertel, 2022; Jospin et al., 2022).

2.2.2.4 Uncertainty estimations by BCNNs

The BCNN used in this study was introduced by Hertel (2022). The work also included estimating two different uncertainty definitions elaborated further in this section. Conventional CNN used for water segmentation generate a deterministic sigmoid output, as described in section 2.2.1. Here a single binary watermask is generated, and the model setup does not change once the model is trained, regardless of how often a prediction is conducted on the input. In contrast, for BCNNs, the setup changes for every prediction resulting in altering results. As a result, numerous predictions create a series of different sigmoid outputs for one prediction input. Based on this series, a probabilistic sigmoid ensemble output is created. This means that for each pixel, multiple sigmoid values are generated. The model is also capable of creating a binary watermask. For this purpose, the mean of all deterministic sigmoid outputs is calculated, and the watermask is created using a threshold. Figure 8 illustrates the difference between deterministic sigmoid output and a probabilistic sigmoid ensemble output.

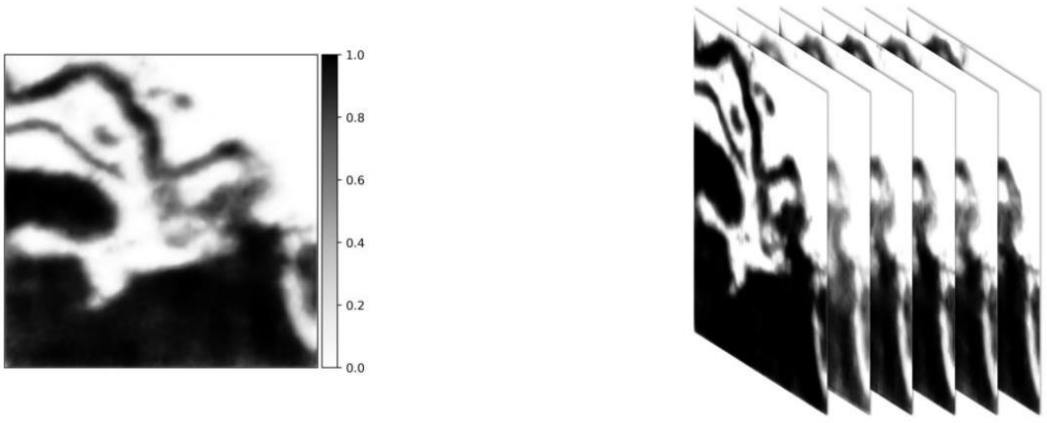


Figure 8: Deterministic sigmoid output (left) and probabilistic sigmoid ensemble output (right) (Hertel, 2022).

The probabilistic sigmoid ensemble output by the BCNN can be utilized to estimate the uncertainty. The uncertainty is often referred to as the spread of the sigmoid values per pixel (Blundell et al., 2015; Hertel, 2022). The uncertainty can be determined by the width of a confidence interval around the mean. This interval span can be expressed as $\mu \pm \sigma$, the uncertainty for each pixel is therefore defined as $(\mu + \sigma) - (\mu + \sigma) = 2\sigma$ (where μ is the mean and σ is the standard deviation). The range of 2σ is no fixed definition and can be adjusted based on the field of application (Hertel, 2022). The definition also

corresponds to approximating the aleatoric uncertainty mentioned in section 2.2.2.1. Figure 9 illustrates the uncertainty as the range between the red line, marking the confidence interval.

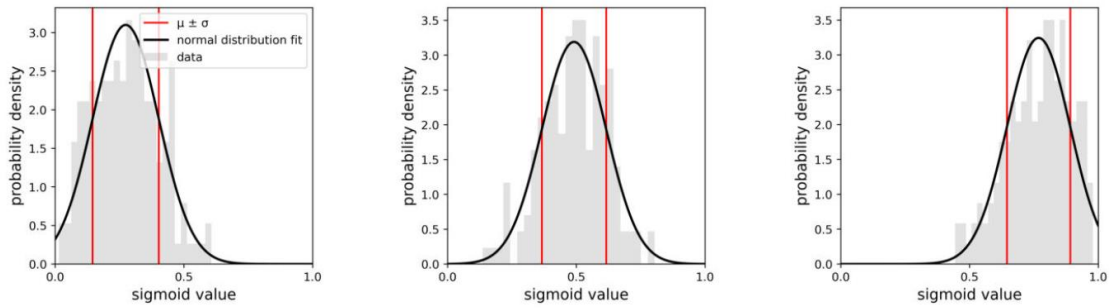


Figure 9: Uncertainty based on the spread of the sigmoid distribution. Displayed for a water pixel (left), an unclear pixel (middle), and a non-water pixel (right) (Hertel, 2022).

This definition of uncertainty does not consider the position of the mean. A pixel in which the sigmoid distribution lies in a close approximation of either 0 (non-water) or 1 (water) might possess the same uncertainty as a pixel with a mean close to 0.5, depicting an unclear pixel. This is also recognizable in Figure 9. To account for this problem and to consider the position of the mean, Hertel (2022) proposes an extension of the uncertainty definition. Here the class probabilities are estimated by the areas under the probability function, left and right, of 0.5. Therefore, uncertainty is the probability of a class being correctly predicted by the BCNN. The probability can take values from 0.5 to 1, 0.5 describing the highest uncertainty of an unclear pixel. The approach is visually presented in Figure 10. The orange and blue coloration indicates the probability as the area under curve for class 0 (orange) and class 1 (blue).

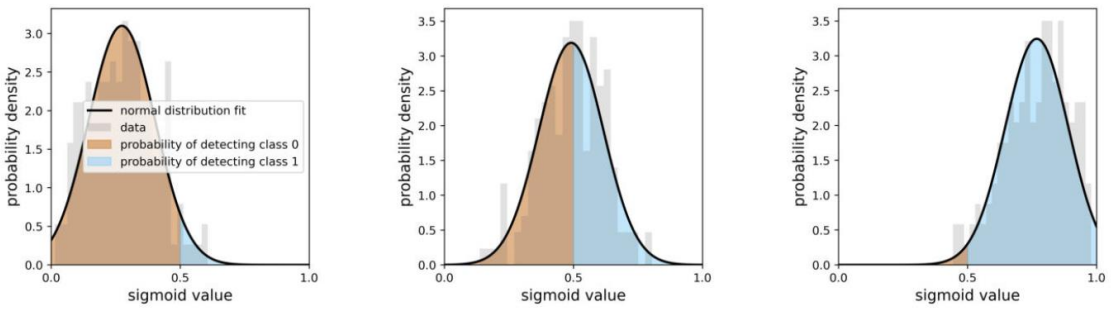


Figure 10: Uncertainty as the probability of a pixel being correctly labeled. Displayed for a water pixel (left), an unclear pixel (middle), and a non-water pixel (right) (Hertel, 2022).

These definitions are further referred to as Uncertainty Definition 1 (UD1: Uncertainty as the confidence interval range) and Uncertainty Definition 2 (UD2: Probability of a class being correctly labeled).

2.2.2.5 Application of BCNNs and Uncertainty Estimations

Implementing the Bayesian approach to CNNs has opened the possibility of comprehensively estimating the uncertainty of a model's prediction. This uncertainty can originate from various sources. For example, the reduction of inaccuracies of CNNs for water detection is limited by the quality and completeness of the training data. This can lead to severe consequences in operational applications, for example, in medical imagery and flood maps provided to the first responders in case of an emergency. Uncertainty estimations might prove an important step to further understanding the models' inaccuracies.

Bayesian uncertainty was first successfully applied to CNNs by Gal & Ghahramani (2015). They approximated the posterior distribution with Bernoulli variational distributions and determined the uncertainty by calculating the variance over multiple predictions. Shridhar et al. (2019) produced uncertainty estimations using variational inference for the posterior approximation. BCNNs have also been used in the field of remote sensing. Landcover classifications have been conducted in hyperspectral data using Bayesian modeling (Haut et al., 2018; Joshaghani et al., 2022). Dera et al. (2020) proposed BayesSAR, a BCNN, to perform image classification in SAR data and retrieve the uncertainty estimations based on the variance of the results. Their analysis confirmed highly reliable uncertainty values generated by the BCNN. The findings suggest that areas of higher uncertainty correlate with wrongly labeled areas. This also corresponds to findings in other studies (Wei & Chen, 2021). The methodology proposed by Hertel (2022) is the first approach to utilize BCNNs for water and flood mapping in SAR imagery. He developed a BCNN and a Monte Carlo Dropout Network to derive watermasks and the corresponding uncertainty estimations from Sentinel-1 data. The study also introduces the UD2 mentioned in Section 2.2.2.4. His thesis aimed to display an approach to integrating uncertainties into map products.

3 Research Objectives

The following section highlights the research gaps that became evident in Chapter 2. Furthermore, the research conducted in this study is described in detail, particularly the research questions that address how to fill the research gaps.

There exists a wide range of methods for water and flood mapping. In most cases, the research aims to extend the applicability of existing knowledge to respond to flood events. Flooding can represent a threat to society and the environment. As mentioned in Chapter 1, there is an increase in the intensity of heavy rainfall events predicted, resulting in more severe flooding. A series of different methods to utilize SAR data for flood mapping has been developed, as described in chapter 2. Especially the advancements in the field of deep learning have led to mapping products that are more quickly generated and more accurate than those created by conventional methods. A recent development has been made in the introduction of BCNNs. By providing uncertainty estimations, additional information is provided. However, very little research has been conducted on how well the models perform globally and what conditions persist in error-prone areas. Also, there is little knowledge about the further applicability of uncertainty estimations. This study aims to close these knowledge gaps. An analysis of the performance of the model is conducted. The behavior of uncertainty values over different preexisting conditions is analyzed. This thesis also introduces a way to improve the water surface classification results in SAR data based on the retrieved uncertainties. This approach follows the findings of Redekop & Chernyavskiy (2021). The study aims to answer the following research questions:

1. *Do misclassified pixels lie within specific classes of land cover?*
2. *Are misclassified pixels correlated with increased uncertainty?*
3. *Do pixels with higher uncertainty tend to lie within certain land cover classes?*
4. *Does changing pixels with a high uncertainty to the opposite binary class improve classification results?*
5. *How do the results differ depending on the chosen uncertainty definition?*

4 Data

This chapter describes the data used in this study. The training data for the BCNN consists of Sentinel-1 data and the corresponding digitized valid or reference water masks (4.1). The Sentinel-1 data contains information about the backscatter intensity for the two polarizations, VV and VH. The matching ground truth mask was obtained by manual digitizing using comparable Sentinel-2 data. This dataset was developed by the Natural Hazards team of the department Geo-Risks and Civil Security at the German Aerospace Centre (Wieland et al., 2022). In addition, a global landcover dataset was obtained to analyze the spatial distribution of wrongly labeled pixels and uncertainty values (4.2).

4.1 Global Sentinel – 1 Dataset and Reference Data

To train a CNN to detect water surfaces on a global scale, a training dataset with globally distributed scenes is required as input. The scenes should cover different climate zones, altitudes, and landcover types. This is necessary for the model to be able to learn and adapt to different conditions and provide the possibility to be deployed in various scenarios. The dataset used to train the BCNN in this study consists of 76 globally distributed Sentinel-1 scenes that are Level-1 IW ground range detected (GRD) (Wieland et al., 2022). The selection of suitable scenes was made using a stratified random sample following Wieland & Martinis (2019) to cover a reasonable number of geographically diverse regions (Helleis et al., 2022; Hertel, 2022). The method is based on a global biomes map with a minimum of 370 km between each scene (Olson et al., 2001).

Sentinel-1 is a radar satellite mission consisting of two satellites orbiting the earth at an altitude of 693 km with a 12-day repeat cycle. Level-1 GRD Sentinel-1 scenes comprise the SAR data projected to ground range by utilizing an earth ellipsoid model (ESA, 2021). The scenes are radiometrically calibrated and geometrically corrected following Twele et al. (2016). By comparing the data to the area of the resolution cell on the ground, backscatter intensity values in SAR images are calibrated. Values describing the Normalized Radar Cross Section (NRCS) are the result of calibration. This transformation makes it possible to characterize different aspects using the backscatter data. Additionally, the

polarization information (VV and VH) is kept and used as additional training inputs (e.g., Hertel, 2022).

For the generation of the corresponding watermasks, additional Sentinel-2 data is gathered. Their acquisition lies within a 30-day range around the acquisition date of the Sentinel-1 scenes. They are used to create ground truth masks. These binary masks are created by applying Otsu's method to the Normalized Difference Water Index (NDWI) (Helleis et al., 2022; Hertel, 2022; Otsu, 1979). The automatically created watermasks are manually checked for errors. This is being done to guarantee the highest possible data quality used for the training. Errors might be caused by the temporal gap between Sentinel-1 and Sentinel-2 data. The pixels are labeled as invalid to compensate for areas where Sentinel-2 data was unreliable. Unreliable pixels may be identified due to cloud cover in a scene or the pixel's proximity to the edges of the scene. Since Sentinel-1 operates independent of weather conditions, invalid pixels are only detected in the Sentinel-2 data.

The dataset comprising 76 Sentinel-1 scenes is preprocessed by being split into 60% training scenes, 20% validation scenes, and 20% test scenes. The splitting happens on a scene level to avoid spatial autocorrelation when the data is split on a sub-level. As mentioned in section 2.2.1, the training data is used for the actual training process, and the model uses the validation data to evaluate the learning progress during the training. Using the test scenes, the performance of the final model to detect the target class can be assessed with data that the model has not seen before. The training and validation scenes are tiled into 256 x 256 pixel tiles. Tiling is applied to the SAR data and the corresponding reference water masks, resulting in 102.676 training tiles and 46.663 validation tiles. Predictions are performed on the whole image of the 18 test scenes. Figure 11 illustrates the global distribution of the scenes and highlights the variable geographic conditions in which they are found.

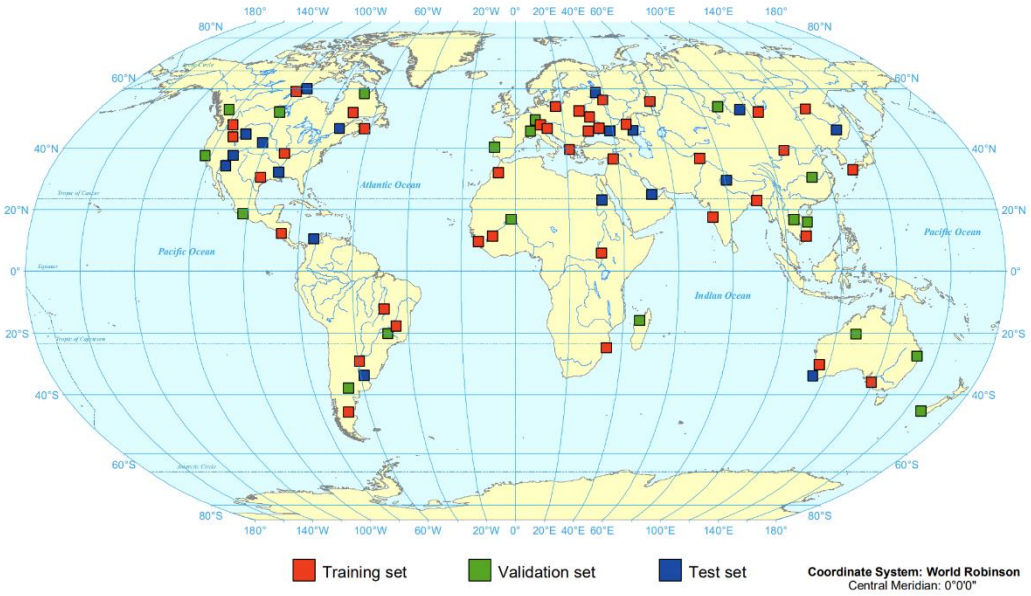


Figure 11: Global distribution of the training, validation, and test Sentinel-1 scenes (Hertel, 2022).

4.2 Copernicus Global Landcover Data

This study conducts experiments to provide insight into the spatial distribution of wrongly labeled pixels and estimated uncertainties generated by the BCNN. The data used for the evaluations is a globally available landcover dataset. Landcover and land use provides an excellent overview of various regional and local conditions. A global dataset also provides comparability between the scenes used to evaluate the model performance. As landcover products need to cover large spatial areas, in-situ measurements are costly and can only produce point estimates that need to be interpolated (Buchhorn et al., 2020). Most landcover data are derived from satellite remote sensing to compensate for this. There exist a variety of available data. Data like the CORINE Land Cover product (for Europe) or the National Land Cover Database (for the United States of America) provide landcover information on a national or continental scale (Bossard et al., 2000; Homer et al., 2020; Rigge et al., 2021). Global landcover products are also available, like the Global Land Survey (GLS) or the Copernicus Global Landcover (CGL) data (Buchhorn et al., 2020; Gutman et al., 2013). Copernicus Global Land Service Land Cover Map at 100m (GCLS-LC100) data is used in this study. GCLS-LC100 is a global landcover product developed by the European Copernicus service with a 100m spatial resolution. The dataset was created for 2015 – 2019 and made available in 3 different collections. The dataset is selected for this study since the existing GCLS-LC100 data matches the years of

acquisition of the Sentinel-1 data. It should be noted that using an annual landcover product raises certain limitations. Landcover is variable over a year. This variability also strongly depends on the climate zone (Allan et al., 2014). The data and the achieved results are to be seen in this context.

The GLCS-LC100 product is created using the PROBA-V sensor. PROBA-V is a satellite constructed by the European Space Agency to provide data that can be used for vegetation monitoring. The input data for the actual landcover classification is not the complete backscatter information but t10 Vegetation Indices that are calculated (Buchhorn et al., 2019, 2020). Additionally, 270 metrics like geomorphological features and descriptive statistics are generated following Zhai et al. (2018) and Eberenz et al. (2016). Since the CLCS-LC100 algorithm does not classify water surfaces and built-up areas, additional data is fused after the initial landcover classification. This data consists of the JRC Global Surface Water dataset and the World Settlement Footprint created by the Germany Aerospace Centre (Buchhorn et al., 2020; Marconcini et al., 2019; Pekel et al., 2016). In further preprocessing, less relevant metrics are dropped to create the optimal dataset for the classification.

The reference data used for model training was collected and created by 20 trained experts via the Geo-Wiki platform (Buchhorn et al., 2020; Fritz et al., 2012). The optimized training data is used as the input of the supervised classifier. First, a Random Forest Classifier is used as the base classifier. Next, a discrete landcover map and information about class probability and vegetation coverage per pixel are created (Buchhorn et al., 2019). In the final step, the result is combined with auxiliary data, consisting of water surface, built-up areas, and snow data. The discrete landcover map consists of 25 different classes. After classification, an extensive accuracy assessment is conducted. Table 2 displays the number of taken samples, the achieved overall accuracy, and the corresponding confidence intervals for all continents. An overall accuracy of 80.2% +/- 0.7% is achieved, the latter being the confidence intervals at 95% confidence levels. Concerning class-specific accuracies, permanent water, bare soil, snow/ice, and forest achieve high accuracies of over 85%. Herbaceous vegetation, built-up areas, and cropland reach moderate accuracies between 70% and 85%. Shrubs, herbaceous wetland, and moss/lichen have lower accuracies of under 65% (Buchhorn et al., 2020). These accuracy assessment results are noted for further analysis in this study as two remote sensing products are involved, both prone to specific errors.

Table 2: Number of taken Accuracy Assessment samples, achieved overall accuracy and corresponding confidence intervals for each continent (Buchhorn et al., 2020).

	Number of Samples	**Overall Accuracy (%)**	**Confidence Intervalls ±**
Africa	3,613	80.1	2.0
Asia	3,071	83.3	1.5
Northern Eurasia	2,976	79.8	1.6
Europe	3,120	80.4	1.6
North America	2,843	77.1	1.7
Oceania & Australia	2,951	81.9	1.9
South America	3,017	79.6	1.5

The GLCS-LC100 data used in this study is retrieved using Google Earth Engine (GEE). GEE is a platform for the easy analysis and acquisition of geospatial data (Gorelick et al., 2017). The data is acquired for all 18 test scenes. All test scenes' extent and resolution are used as additional parameters for the GEE download. As the extents between the Sentinel-1 scenes and the GLCS-LC100 data do not perfectly align, a clipping of the data is performed. For further analysis, the data must be perfectly matched. Some of the 22 initial classes are summarized to clarify the results better. In detail, the 12 different forest classes are summarized into the classes open forest and closed forest. This leads to a total of 12 landcover classes. Their definitions are outlined in Table 3.

Table 3: Summarized GLCS-LC100 landcover classes used in this study (changed following Buchhorn et al. (2020)).

Land Cover Class	**Definition**
Shrubs	These are woody perennial plants with persistent and woody stems without any defined main stem being less than 5 m tall. The shrub foliage can be either evergreen or deciduous.
Herbaceous Vegetation	Plants without persistent stems or shoots above ground and lacking definite firm structure. Tree and shrub cover is less than 10 %.
Cultivated Vegetation	Lands covered with temporary crops followed by harvest and a bare soil period (e.g., single and multiple cropping systems). Note that perennial woody crops will be classified as the appropriate forest or shrub land cover type.
Urban and Built-up	Land covered by buildings and other man-made structures.
Bare Soil	Lands with exposed soil, sand, or rocks and never has more than 10 % vegetated cover during any time of the year.
Snow and Ice	Lands under snow or ice cover throughout the year.
Permanent Water Bodies	Lakes, reservoirs, and rivers. Either fresh or salt-water bodies.
Moss and Lichen	Moss and lichen.
Herbaceous Wetland	Lands with a permanent mixture of water and herbaceous or woody vegetation. The vegetation can be present in either salt, brackish, or fresh water.
Closed Forest	Tree canopy >70 %, mix of closed forest types.
Open Forest	Tree canopy 15-70 %, mix of open forest types.
Seas	Oceans, seas. Can be either fresh or salt-water bodies.

5 Methodology

The following chapter describes the methodology of the experiments conducted in this study. First, the setup of the used BCNN is explained (5.1). Then, it will be addressed which CNN architecture is being used. Furthermore, the hyperparameters used for the model are outlined. Next, the uncertainty definitions and the methodology used to quantify the uncertainty are illustrated. This uncertainty analysis is performed for the correctly and wrongly labeled pixels and the different landcover classes of the GLCS-LC100 data (5.2). Next, a methodology is introduced to use uncertainty estimations to optimize the results obtained from the BCNN (5.3). Finally, this chapter's last section provides an overview of the trained models and the performance metrics used to evaluate the BCNNs predictions (5.4 and 5.5).

The experiments were conducted using Python scripts. A static version of the code base is provided as a repository under: https://gitfront.io/r/user-1618937/HXJvhZE2zo6c/mamederer/. For access to the GitHub repository please contact the author.

5.1 Setup of the Bayesian Convolutional Neural Network

This section outlines the setup of the used BCNN. The BCNN utilized in this study was first introduced by Hertel (2022). The BCNN has a modified U-Net architecture. The U-Net architecture is displayed in Figure 12.

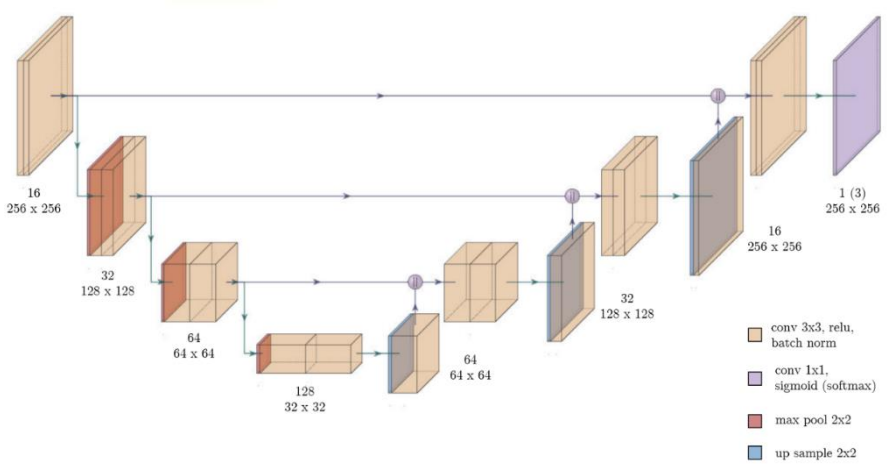


Figure 12: CNN U-Net architecture (Hertel, 2022).

The dimensions of the convolutional layers are first reduced by a 2 x 2 max pooling and later increased by 2 x 2 upsampling. For BCNNs, the weights or filter values of the convolution are not fixed values but rather probability distributions. These act as the prior distribution for the Bayesian inference approach. The convolutional layers are initialized with a standard normal prior distribution $P(w) = N(0,1)$, where w describes the weight. The distribution is approximated during the forward passes using a flip-out estimator (Hertel, 2022; LaBonte et al., 2019; Wen et al., 2018). The loss function is described as the Evidence Lower Bound (ELBO). It is a combination of cross-entropy and KL-divergence (Hertel, 2022). The hyperparameters visible in Table 4 are used for the model's setup. A detailed explanation of all parameters can be found in section 2.2.1.5.

Table 4: Hyperparameters used for the setup of the BCNN (changed following Hertel (2022)).

Parameter	Setup
Initial learning rate	0.0001
Learning rate decay	0.5 / 7 epochs
Maximum epochs	100
Early stopping	10
Batch size	16
Prior distribution	$N\,(0,1)$

This setup is used for two of the three trained models. The reasoning for this and the other setups can be found in sections 5.3 and 5.4. The predictions are conducted 32 times to create a probabilistic sigmoid ensemble output. Based on this output, the uncertainties are estimated, extracting the uncertainty definitions 1 and 2. The uncertainties are used for further analysis in this study. The model was created using the Python TensorFlow framework (Abadi et al., 2016; Ghemawat et al., 2016; Hertel, 2022; vanRossum, 1995).

5.2 Identification of error-prone regions of the BCNN

There are multiple error sources when extracting water surface areas from SAR data. Over- and underestimations can happen when surfaces alter the backscatter, as explained in section 2.1.3. No experiments have been conducted to detect error-prone areas of water detection in SAR data using a BCNN. This study introduces a method to quantify the

proportion of wrongly labeled pixels over different types of prevailing conditions. The experiment uses the binary segmentation result of the BCNN and the GLCS-LC100 landcover data. The binary water masks are derived by thresholding the mean sigmoid values from all 32 predictions made by the BCNN. The proportion $P(cw)$ of all wrongly labeled pixels lying within one landcover class is calculated by:

$$P(cw) = \frac{cwp}{wp}$$

Where

cwp Number of misclassified pixels in a class

wp Total number of misclassified pixels in all classes

This experiment is conducted to detect the error-prone areas for water detection in SAR data mentioned in the literature. In addition, this methodology aims to answer whether misclassified pixels tend to lie within certain landcover classes.

5.3 Uncertainty Analysis

The BCNN developed by Hertel (2022) produces uncertainty estimations next to the binary output. As mentioned in section 2.2.2.4, the model produces uncertainties based on two definitions, representing different interpretations of the probabilistic ensemble output. The prediction of the input for one model is conducted 32 times, as mentioned in 5.1. For every pixel, 32 sigmoid outputs are generated. This is utilized to determine the uncertainties. Definition 1 is the range of ± 1 standard deviation around the mean. Definition 2 also takes the position of the mean into account. It is defined as the probability of a class being correctly labeled. Definition 1 can take values between 0 and 1, with 1 presenting the highest uncertainty. The probabilities of Definition 2 can take values from 0.5 to 1, with 0.5 constituting the lowest probability and, thus, the highest uncertainty. The uncertainty analysis conducted in this experiment compares the results of the different models that are trained (see section 5.4) as well as the two uncertainty definitions. The uncertainty definitions are subsequently referred to as UD1 (spread of values) and UD2 (spread and position of mean).

5.3.1 Uncertainty Distribution over correctly and wrongly labeled pixels

Two different types of uncertainty analyses are conducted in this study. The first aims to answer the research question of whether misclassified pixels correspond to increased uncertainty values compared to correctly labeled pixels. This is also a primary hypothesis for the methodology introduced in section 5.3 to improve the classification results based on the obtained uncertainty estimations. Quantitative and qualitative diagnostics are conducted in this experiment. Prior to the analysis, the prediction results are split into pixels that are True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN). The four classes are commonly used for the accuracy assessment of remote sensing products (Foody, 2002; Story & Congalton, 1986). This separation of cases is done by comparing the prediction water mask to the reference mask created with the methodology described in section 4.1. The statistics of the uncertainty values are calculated. The mean, median, Quartile 1 (Q_1), Quartile 3 (Q_3) as well as the Interquartile range (IQR) are obtained. Q_1 and Q_3 are representing the 25th and 75th percentiles of the data, respectively. The IQR is a measure of statistical dispersion used to gain insights into the spread of the data. It is defined as the difference between Q_3 and Q_1, $IQR = Q_3 - Q_1$. Boxplots are created using Python and the Matplotlib library to visualize this information. They display the median, Q_1 and Q_3 as the bounding boxes and the whiskers as 1.5 times the IQR from both bounding box boundaries (Q_1 and Q_3) (McGill et al., 1978). The width of the box is set to present the proportion of the class in the complete data. The boxplots visually present the uncertainty distributions over the areas containing TP, FP, TN, and FN pixels.
To further visualize the uncertainties over misclassified and correctly classified areas, map products are generated to provide information about the spatial distribution of the uncertain areas. The maps contain the Sentinel-1 data, the reference water mask, and the corresponding Sentinel-2 data to allow a visual, qualitative interpretation of the location of highly uncertain areas. The figures also display maps of the binary prediction result of the model and the corresponding uncertainty values. To provide further visual information about the uncertainties, the pixels are classified to resemble the classes Accurate and Certain (AC), Accurate and Uncertain (AU), Inaccurate and Certain (IC), and Inaccurate and Uncertain (IU). This is done following the approach by Hertel (2022). Pixels are labeled as uncertain when the uncertainty values exceed 0.1 for UD1 and when they fall below 0.95 for UD2.

It should be noted that the generated map results may vary heavily based on the chosen threshold. The threshold is chosen based on the findings of Hertel (2022) and from visual interpretations of the results of this experiment. The boxplots were created using Python and the Matplotlib library, and the maps were created in QGIS (Hunter, 2007; QGIS Development Team, 2022; vanRossum, 1995).

5.3.2 Uncertainty Distribution over different types of landcover

The second part of the uncertainty analysis returns uncertainty distributions over different underlying land cover conditions. Similar to the approach in 5.2, the experiment is conducted based on correlating the GLCS-LC100 landcover data with the prediction results of the BCNN. Here, the uncertainty values over the different landcover classes are analyzed. For this purpose, the descriptive statistics mean, median, Q_1, Q_3, and IQR of the uncertainty values are calculated over each land cover type. Boxplots, structured as mentioned above, are created for better visualization. The uncertainty analysis is conducted for both uncertainty definitions.

5.4 Result optimization

Uncertainty estimations have been used to provide additional information about the performance of a BCNN and its limitations. This section introduces a method developed by Redekop & Chernyavskiy (2021). This thesis implements their approach for the first time for water detection in SAR data and serves as an extension of the initial workflow.
Training data for the training of CNNs and BCNNs suffer from label noise. Label noise describes inaccuracies in the creation of reference data that is used in the training process. The noise can have multiple origins. The primary sources of the label noise are human-made mistakes, inter-observer variability due to human subjectivity, and errors in the automatic generation of reference data (Redekop & Chernyavskiy, 2021). There have been attempts to compensate for label noise by reweighting the image (Foody, 2002; Ren et al., 2018). As there is an increased number of methods, a reduction of label noise has been achieved by presenting the areas of high uncertainty to an expert for relabeling. This approach can also be made automatically, as Köhler et al. (2019) introduced. They used the produced uncertainty estimations to detect and remove noisy labels iteratively. This

approach was extended by Redekop & Chernyavkiy (2021) and applied to binary segmentation results. They improved the binary classification results by implementing the information from the uncertainty estimations. In their study, they created noisy data. This data was then used to train the Deep Learning model and retrieve the binary predictions and uncertainty estimations. Areas of high uncertainty were then relabeled to their opposite binary class. The primary hypothesis of this approach is that areas of high uncertainty are misclassified. This label improvement possesses the advantage that the reference data does not have to be absolutely accurate. This could save time and resources when creating training data while guaranteeing high model accuracy (Redekop & Chernyavskiy, 2021). Their approach is modified and tested for the applicability of optimizing watermasks created using SAR data and a BCNN.

5.4.1 Creation of label noise

To test their approach, Redekop & Chernyavskiy (2021) generated artificial noise in their training data. They used a low-vertex polygon approximation to introduce noise. This approach is also tested in this study. To create the noisy labels, the Visvalingam-Whyatt algorithm was used (Visvalingam & Whyatt, 1993). The algorithm is used for line simplification by removing vertex points in a line. This is achieved by a technique called effective area for progressive simplification of lines. The effective area is calculated by forming a triangle between three consecutive line points. The smallest area between three points along the line is compared to a set threshold. The point without adjacent points is removed if the area lies below that threshold. The test is then repeated with the second largest triangle. This process is repeated until all triangles under the threshold are eliminated. Figure 13 illustrates the point removal and the resulting simplification of the line. The algorithm was chosen based on its simplicity and good performance. It was applied to the whole training and validation dataset. This provides comparability of the results of the models trained with the simplified and original data.

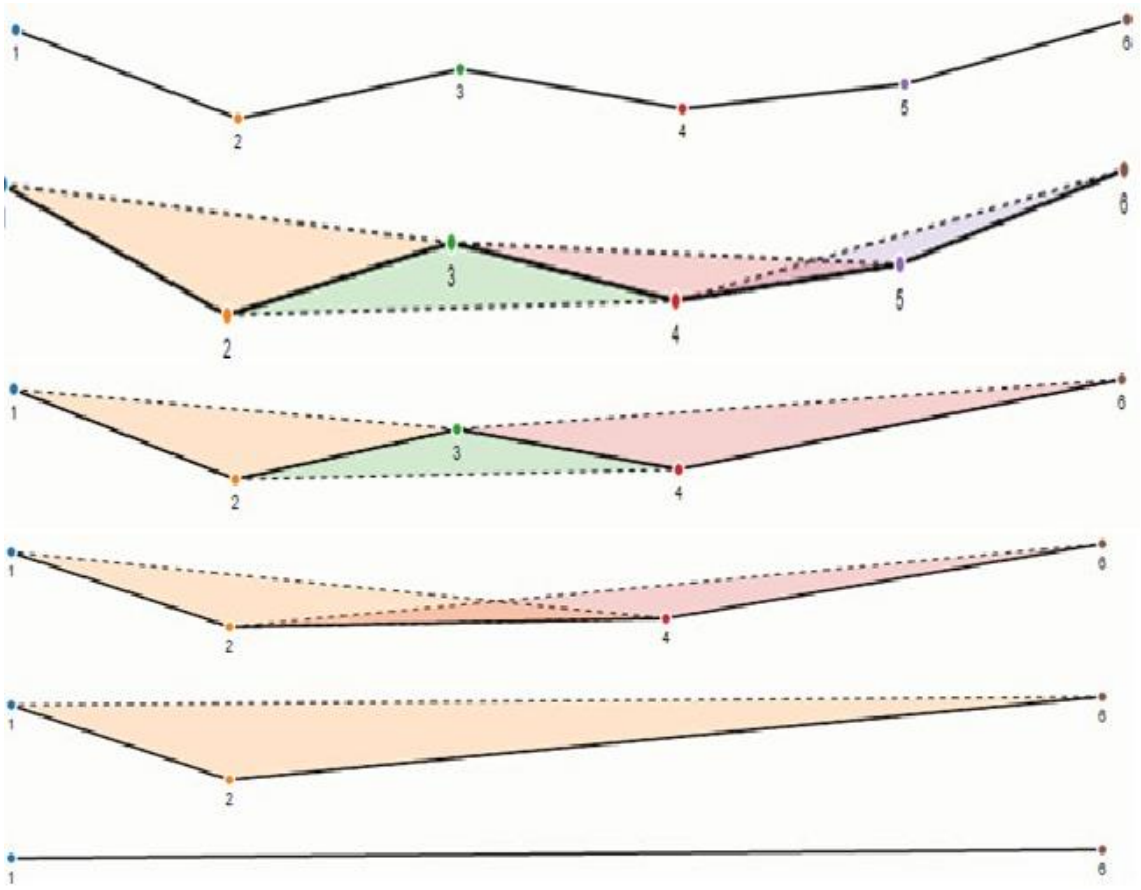


Figure 13: Line simplification using the Visvalingam-Whyattt algorithm (Melnyk & Shokur, 2016).

The Visvalingam-Whyatt algorithm was applied using QGIS, and the tolerance Parameter establishing the area threshold was set to 200 m^2.

5.4.2 Uncertainty estimation and Morphological filtering

In their study, Redekop & Chernyavskiy (2021) trained their model for 30 epochs to determine their approach's optimal training duration. They determined this as the epoch where the velocity at which the mean cumulative uncertainty decreases reaches its highest point. This point is marked as the point where the uncertainty over the wrongly labeled regions is the highest. Therefore, this would be the point where the optimal data for the relabeling process would be generated. This was the case after just one and two epochs of training. According to their findings, the two different BCNNs are trained using the simplified data as training and validation data. According to the findings of Redekop & Chernyvskiy (2021), one model is trained for just one epoch, and the other is trained for

a maximum of 100 epochs with an early stopping set at ten epochs. This is done to confirm whether the shorter training is more suitable for this application.
Before relabeling the predictions, a mask of the pixel positions to be relabeled is created. All pixels above or below (depending on Uncertainty Definition 1 or 2, respectively) a chosen uncertainty threshold were selected, and a binary relabeling mask was created. The relabeling mask determines the position of the pixels that are relabeled. During the experiment, it was discovered that there are overestimations of highly uncertain pixels in the border regions of the mask. This was determined visually and by trial-and-error. The findings mean that there are pixels that would be relabeled to the false label that can be found in a pattern around regions that are IU and should be relabeled. Maps visualizing this phenomenon can be found in section 6.3 and Appendix B. Therefore, it is assumed that shrinking the area of the relabeled pixels might improve the results further. For this purpose, morphological erosion filtering was applied to the relabeling mask before relabeling the data (Heijmans & Ronse, 1990). Erosion is one of the two basic morphological filters, the other being dilation. The purpose of erosion is to shrink the size of objects in a binary image. Objects are pixels that contain a value of 1. The reduction does not only apply to foreground pixels at the outer border, but it also leads to an enlargement of existing holes in the object. The erosion operator takes two inputs, one of which is the binary input. The other one is a filter kernel consisting of 0 and 1. This filter is called the structuring element and impacts the result of the morphological operation. The filter is applied to all pixels from the top left to the bottom right, comparable to the convolutional filter. The difference lies in the mathematical operation. When the filter is applied to a pixel, it checks whether all surrounding pixels lying beneath the filter are foreground pixels (value in the binary mask = 1). If the structuring element fits entirely into the surrounding pixels, the value is left as it is, else it is set to 0 (Dorst & van den Boomgaard, 1994; Heijmans & Ronse, 1990; Schavemaker et al., 2000). In this study, different filter kernels are tested to check their impact on the results of the relabeling.

5.4.3 Relabeling of the generated masks

Highly uncertain areas are detected by selecting pixels that exceed or fall below a chosen uncertainty threshold (depending on the uncertainty definition). This so-called relabeling mask determines the positions of pixels to be relabeled. For the two uncertainty definitions, the creation of the relabeling masks follows the rule:

$$RM_{Unc1} = U > \delta$$
$$RM_{Unc2} = P < \delta$$

Where

RM is the extracted relabeling mask.

U is the array containing the uncertainties for Uncertainty Definition 1.

P is the array containing the probabilities for Uncertainty Definition 2.

δ is the chosen uncertainty threshold above which the pixel shall be relabeled.

After the generation of the relabeling mask, the erosion filter is applied. Then the prediction data is changed according to:

$$p_{new} = 1 - p_{old} \quad \text{, if} \quad RM_p = 1$$

Where

p_{old} is the pixel value in the prediction water mask.

p_{new} is the pixel value after the relabeling step.

Figure 14 provides further visual details about the relabeling step. It should be noted that the erosion filter is applied to the areas of high uncertainty between the detection of the highly uncertain areas and the relabeling.

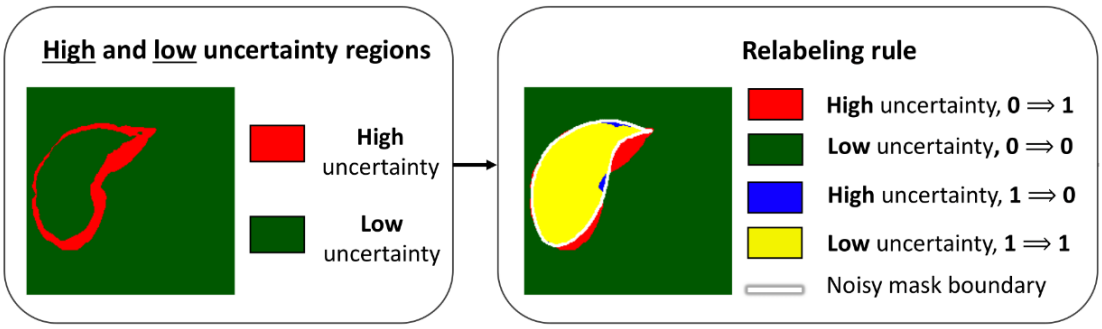


Figure 14: Details of the relabeling step (changed following (Redekop & Chernyavskiy, 2021).

To evaluate the functionality of the proposed method, different thresholds are tested. The experiment also evaluates the effect of three different kernel sizes on the erosion of the relabeling mask. After each parameter adjustment, the changes to performance (see

section 5.5) are monitored. The proposed methodology was programmed using Python and visualized using Matplotlib for graphs and QGIS for map products (Hunter, 2007; QGIS Development Team, 2022; vanRossum, 1995). The method aims to improve prediction accuracies even when the training data contains label noise. This could provide an approach to investing less time and resources into generating ground truth data for model training. It could also provide the opportunity to create reference data over more parts of the world. This could support discussion about the generalizability of classification results over regions that have previously not been integrated.

5.5 Trained Models

This section provides a summary of the BCNNs trained in this study. In total, three different models have been trained and evaluated. The first one (Model A) is trained using the best available training and validation data. It utilizes the setup displayed in Table 4. The Sentinel-1 data and the reference data, created by NDWI and expert knowledge, described in section 4.1, are used for the training process. All 102.676 training tiles and 46.663 validation tiles are used.

The second model (Model B) is also trained using the setup illustrated in Table 4. The difference lies in the training and validation data. The Visvalingam-Whyatt algorithm introduces artificial noise into the data (Visvalingam & Whyatt, 1993). As a result, the 102.676 training and 46.663 validation tiles used for training and validation possess a simplified geometry.

The final model (Model C) is trained using the simplified geometry training and validation data. The difference to the second model lies in the setup. Instead of a maximum of 100 epochs training with ten epochs early stopping, the model is trained for just one epoch. This is following the findings of Redekop & Chernyavsik (2021).

Each model was predicted twice using the 18 test scenes, previously not shown to the BCNNs. Then, inside each prediction, the test scenes are analyzed 32 times to generate the probabilistic sigmoid output for each of the six predictions. Finally, the two predictions for each model were conducted to produce the uncertainties for both definitions.

5.6 Performance Metrics

Water surface detection in SAR data is prone to numerous error sources. This is the case when using conventional and recent Deep Learning approaches. Those errors can occur through human or technological inaccuracies. Thus, differences between detection results and the ground truth are expected and tolerated to an acceptable degree. Multiple performance metrics were developed to detect those erroneous regions and quantify the accuracy of the created masks. The overall accuracy of the classifier identifies the fraction of correctly labeled pixels. However, this can lead to wrong conclusions for imbalanced datasets. For example, if a scene only contains a small number of water pixels, the overall accuracy might be very high since most of the non-water class was correctly labeled. Therefore, reporting this metric alone would fail to capture a low accuracy associated with detecting the target water class.

To compensate for this problem, metrics like Precision, Recall, and the F1-Score were also included as a part of a set of performance metrics. Table 5 provides an overview of the performance metrics used in this study. The Intersection-Over-Union (IoU) further describes the intersection ratio between the prediction and reference water mask. It is a measure of how well the masks match each other. Lastly, Cohen's Kappa (κ) is calculated. The Kappa coefficient shows the difference between the actual agreement and the random agreement between prediction and reference data. All metrics refer to higher performance the closer they are to 1. The metrics were chosen based on suggestions by Hertel (2022), Cohen (1960), and Sokolova & Lapalme (2009).

The performance metrics are first calculated to evaluate the three trained models. Then, f the relabeling process, the metrics and the change to the initial performance are calculated and visualized by line plots generated using Python and the Matplotlib library (Hunter, 2007; vanRossum, 1995).

Table 5: Quantitative metrics to evaluate classification performance (Sokolova & Lapalme, 2009).

Metric	Equation	Description
Accuracy	$\frac{tp + tn}{tp + tn + fp + fn}$	Measures the overall accuracy of a classifier. Can be misleading for imbalanced datasets.
Precision	$\frac{tp}{tp + fp}$	Fraction of correctly detected water pixels.
Recall	$\frac{tp}{tp + fn}$	Fraction of detected water pixels compared to all water pixels.
F1 Score	$\frac{2 \cdot precision \cdot recall}{precision + recall}$	Harmonic mean of precision and recall.
IoU	$\frac{tp}{tp + fp + fn}$	Intersection-over-union (IoU) measures the ratio of the intersection between the reference mask and the prediction mask over their union.
κ	Cohen's Kappa (Cohen, 1960)	Compares the classification result to one achieved by completely random classification.

6 Results

In the following chapter, the results of the conducted experiments are presented. This is done quantitatively and qualitatively, depending on the research question. First, the learning behavior of the trained BCNNs and the initial prediction results for the three models are evaluated (6.1). Furthermore, the error-prone areas of the models are investigated (6.2). Next, the retrieved uncertainty estimations are analyzed. Their spatial distribution over misclassified pixels and different land cover types is presented (6.3). Lastly, the approach to relabel the highly uncertain pixels is tested to improve the prediction results (6.4).

6.1 Initial Prediction Results

This section examines the learning behavior of the trained BCNNs and presents the initial prediction results. Three different models were trained, as described in section 5.5.

Figure 15 provides insight into the learning behavior of the models. The training and validation losses (right y-axis) and accuracies (left y-axis) are displayed over the epochs (x-axis). Models A and B performed well during training, as visible through the training and validation loss curves. They decrease steeply until reaching a plateau after five to seven epochs. Good performance also becomes evident when looking at the training and evaluation accuracies. They peak after one to three training epochs and do not improve with further training. The early stopping was activated after 12 epochs of training for model A and 13 epochs for model B. For model C, training and validation losses and accuracies are only single-point values since the model was only trained for one epoch. The training loss after one epoch was 4.64, and the validation loss 4.25. The training accuracy was 0.93, and the validation accuracy was 0.94. Thus, model C reached a worse training behavior than the other two trained models.

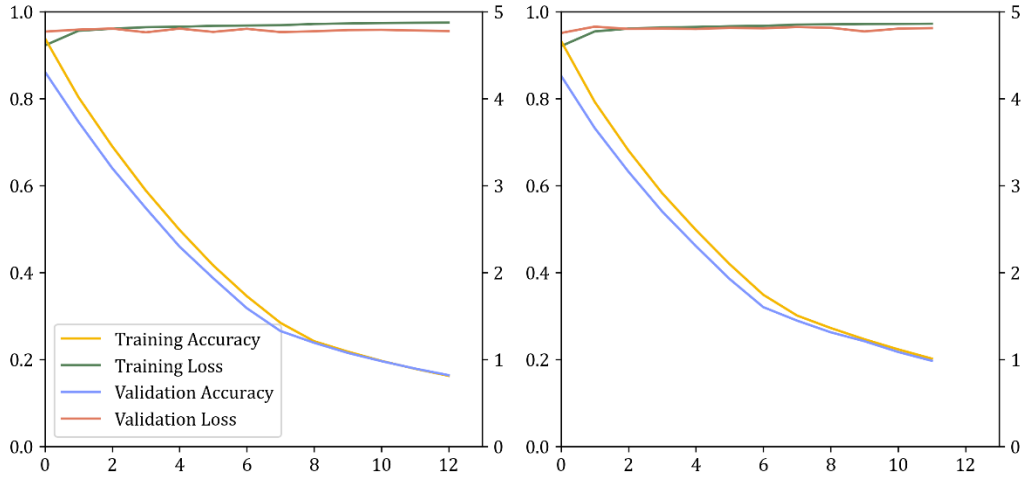


Figure 15: Training and validation logs for the BCNNs trained with the whole dataset (left) and the simplified data (right). The left y-axis represents the accuracy, and the right y-axis the loss. The x-axis is the number of epochs training.

This result becomes more evident when inspecting the initial prediction results for the three models, as shown in Table 6. For each model, two predictions were conducted. This was done to extract the uncertainty values for both uncertainty definitions, leading to a total of six predictions. They are subsequently referred to by their prediction ID stated in Table 6, ranging from 001 to 006.

Table 6: Initial prediction results for the six predictions using the three models.

Prediction ID	Model	Unc. Def.	Acc.	Recall	Precision	F1 Score	IoU	Kappa
001	A	1	0.947	0.775	0.735	0.705	0.618	0.672
002	A	2	0.947	0.775	0.735	0.705	0.618	0.672
003	B	1	0.955	0.709	0.744	0.683	0.590	0.655
004	B	2	0.955	0.710	0.744	0.683	0.590	0.655
005	C	1	0.932	0.729	0.687	0.653	0.575	0.623
006	C	2	0.932	0.729	0.686	0.653	0.574	0.623

The performance metrics show the higher performance of model A. The models trained with the simplified reference generated less optimal prediction results, across all the metrics, except for overall accuracy. Table 6 shows only slight differences in performance metrics between the two uncertainty definitions for each model. To better visualize the result between the three models, the initial prediction results are also displayed in Figure 16. Since the difference between UD1 and UD2 is minimal, only the results for UD1 are

plotted in the figure. It is visible that there is a step-like decrease from model A to model C for the F1-Score, the IoU and the Kappa coefficient.

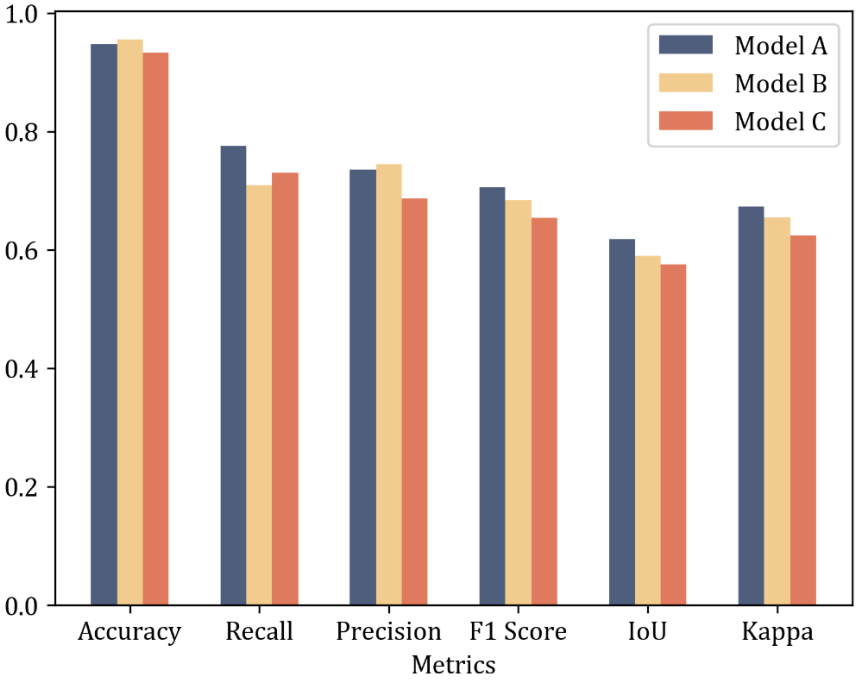


Figure 16: Initial prediction results for the trained BCNNs, trained and validated using the optimal data (Model A), the simplified data (Model B), and the simplified data with one epoch training (Model C).

6.2 Detection of error-prone areas

All six predictions were generated over the 18 test scenes. The scenes were previously excluded from the training process and represent a range of geographic conditions. The experiment regarding detecting error-prone regions was conducted to discover which preexisting factors impact the performance of the BCNNs. Table 7 provides an overview of the 12 landcover classes derived from the GLC-LC100 data over all 18 test scenes. Closed Forest is the most commonly occurring class representing 30.60% of all valid pixels, followed by Herbaceous Vegetation with 16.31% and Cultivated Vegetation with 13.82%. The Snow / Ice and Moss / Lichen classes are notably underrepresented (< 0.05).

Table 7: Overview of the proportion of pixels within each landcover class for the 18 test scenes.

Landcover Class	Pixel Count	Proportion
Shrubs	148,114,596	4.95%
Herbaceous Vegetation	488,060,993	16.31%
Cultivated Vegetation	413,691,605	13.82%
Urban / Built-up	23,474,590	0.78%
Bare Soil	236,802,499	7.91%
Snow / Ice	383,429	0.01%
Permanent Water bodies	216,703,604	7.24%
Herbaceous Wetland	119,088,128	3.98%
Moss / Lichen	537,425	0.02%
Closed Forest	915,991,775	30.60%
Open Forest	281,569,594	9.41%
Seas	148,775,117	4.97%

To provide information about the landcover classes and how they relate to the Sentinel-1 data the predictions are based on, the backscatter values over different landcover classes were analyzed. Figure 17 shows a boxplot for the backscatter values of all 18 test scenes over the 12 landcover classes derived from GCLS-LC100. The low backscatter values over open water, the assumption on which the water surface is extracted, are detectable. There are also lower backscatter values over the classes Bare Soil and Herbaceous Vegetation.

To detect the error-prone areas in the results, the proportion of misclassified pixels within each landcover class was calculated concerning the total number of misclassified pixels. The results of this analysis are displayed in Table 8. For all three models, the highest percentage lies within the class Bare soil. The models trained with the simplified data have a higher error rate than model A which was trained with the optimal reference data. The class Herbaceous Vegetation was also correlated with high proportions of misclassified pixels. Here the highest percentage is detected in results generated with model C. Similar to the findings in section 6.1, it is noted that there is only a minimal difference between the results generated with the two different uncertainty definitions. Thus, the table only presents the results for UD1 of each model.

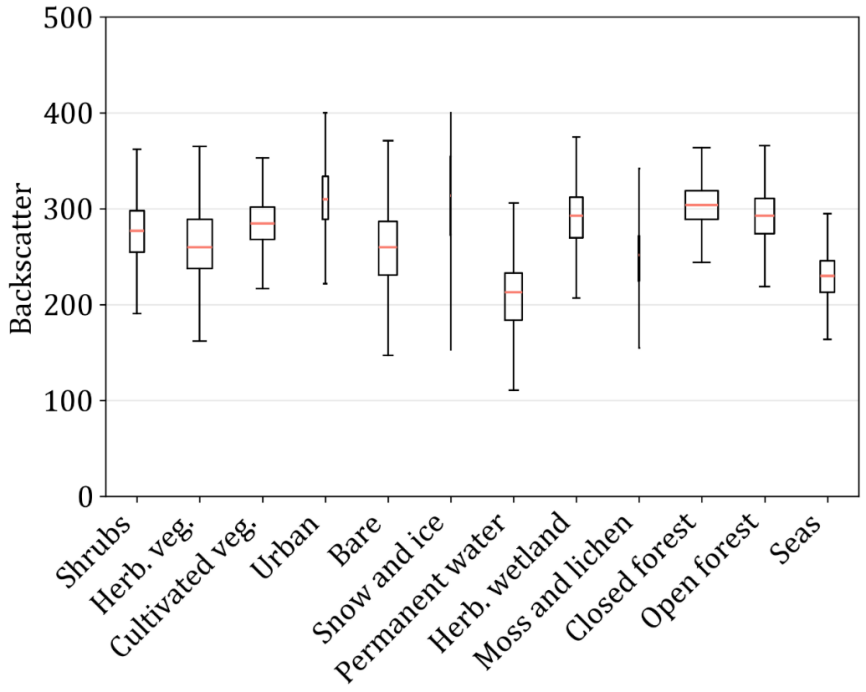


Figure 17: Backscatter values over the 12 different landcover classes for all 18 test scenes.

Table 8: Proportion of misclassified pixels over each landcover class to all misclassified pixels over all 18 test scenes and for all six predictions.

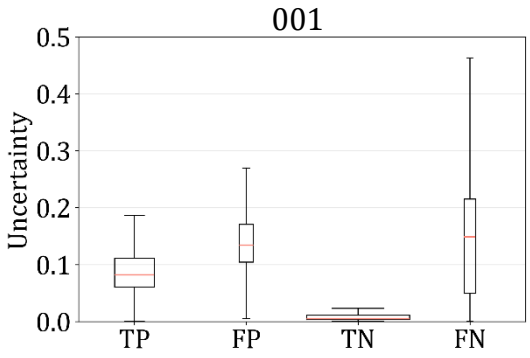
Landcover Class	Model A	Model B	Model C
Shrubs	1.89%	1.61%	2.65%
Herbaceous Vegetation	**26.63%**	**16.58%**	**43.10%**
Cultivated Vegetation	5.57%	5.71%	4.95%
Urban / Built-up	0.09%	0.09%	0.05%
Bare Soil	**30.90%**	**34.40%**	**35.21%**
Snow / Ice	0.00%	0.00%	0.00%
Permanent Water bodies	13.31%	12.11%	5.11%
Herbaceous Wetland	7.57%	6.77%	3.77%
Moss / Lichen	0.39%	0.31%	0.15%
Closed Forest	2.17%	2.16%	1.13%
Open Forest	3.39%	3.65%	2.09%
Seas	8.09%	16.61%	1.79%

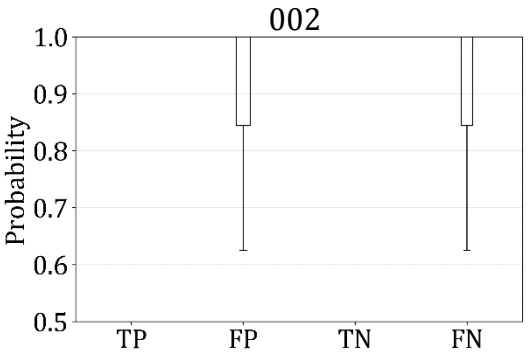
6.3 Uncertainty Analysis

The following section presents the results of experiments regarding the spatial distribution of the uncertainties. This is done to provide a knowledge basis on which further analysis involving the uncertainties can be conducted. The uncertainties were derived based on the spread of the predictions per pixel. Each of the six predictions was made 32 times for all 18 test scenes to generate a probabilistic sigmoid ensemble, thus leading to 32 sigmoid values for each pixel. The uncertainty was derived based on two definitions, as further explained in section 2.2.2.4.

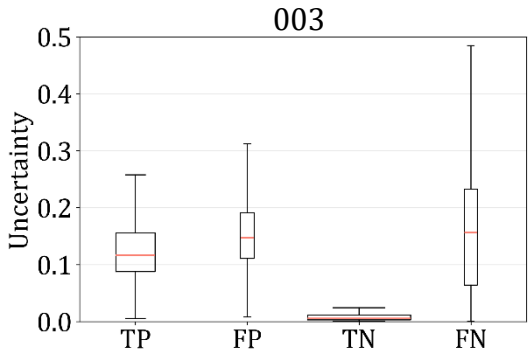
6.3.1 Uncertainty values over misclassified pixels

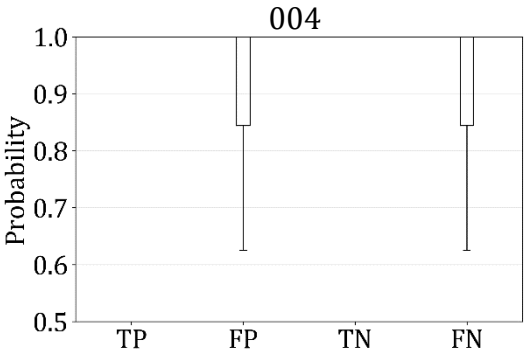
The first experiment regarding the uncertainty estimation regards their spatial distribution over correctly and misclassified pixels. Therefore, the TP, FP, TN, and FN pixels were calculated for all predictions. This was done by comparing the predictions with the reference water mask to evaluate the uncertainty behavior over the four classes. To quantify the values, the statistics mean, median, Q_1, Q_3 and IQR were calculated for the uncertainties over TP, FP, TN, and FN pixels. Additionally, boxplots were created for better visualization of the distributions. Figure 18 illustrates the statistics in the form of tables and the corresponding boxplots for all six predictions. The statistics for UD1 are displayed on the left, and the ones for UD2 are on the right. For all three predictions that generated the uncertainty based on UD1, the mean, median, Q_1, and Q_3 pose a higher uncertainty over misclassified pixels (FP, FN). This confirms the hypothesis that there are higher uncertainties detected over misclassified pixels. For UD2, the mean and median also present a higher uncertainty over misclassified areas. When looking at the IQR to measure the variability of the data, it is detectable that the IQR is higher over falsely classified pixels. This is the case for all six predictions independent of the estimated uncertainty definition. Looking at the boxplots and statistics, it becomes apparent that the uncertainties calculated using UD2 have low variability, close to 1. This becomes evident as the median is 1 for all three models.



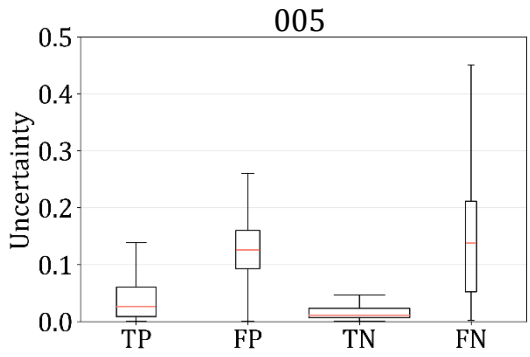


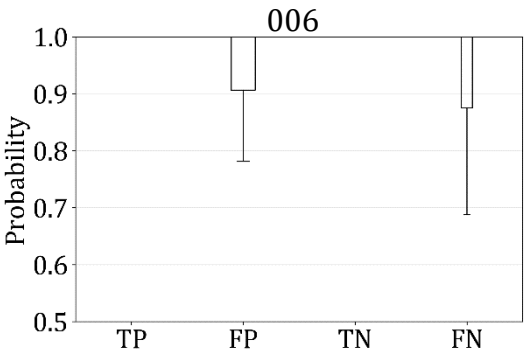
	TP	FP	TN	FN	TP	FP	TN	FN
Mean	0.091	0.143	0.019	0.145	0.985	0.901	0.997	0.902
Median	0.082	0.134	0.005	0.145	1	1	1	1
Q_1	0.061	0.104	0.003	0.045	1	0.844	1	0.844
Q_3	0.111	0.170	0.011	0.215	1	1	1	1
IQR	0.050	0.066	0.008	0.165	0	0.156	0	0.156





	TP	FP	TN	FN	TP	FP	TN	FN
Mean	0.129	0.157	0.018	0.164	0.981	0.902	0.997	0.909
Median	0.116	0.147	0.005	0.156	1	1	1	1
Q_1	0.088	0.111	0.003	0.064	1	0.844	1	0.844
Q_3	0.156	0.191	0.011	0.232	1	1	1	1
IQR	0.068	0.081	0.009	0.168	0	0.157	0	0.156



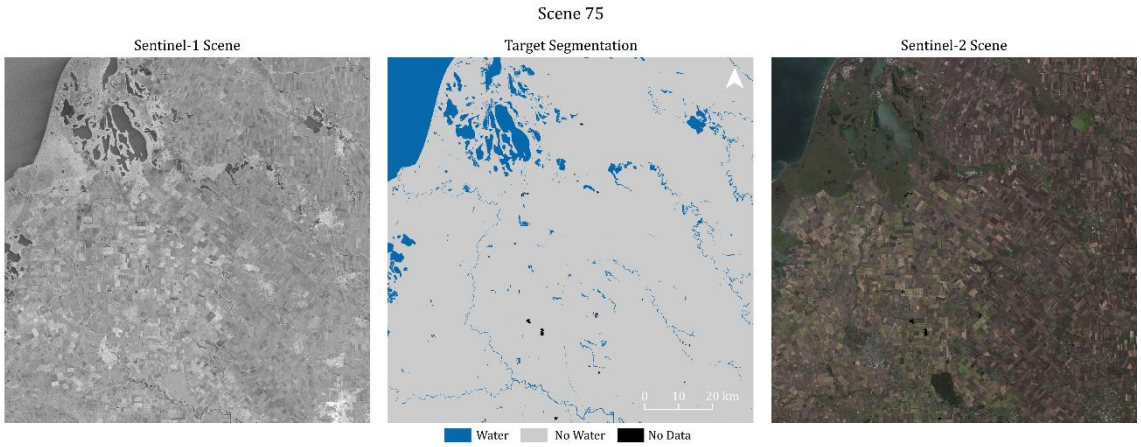


	TP	FP	TN	FN	TP	FP	TN	FN
Mean	0.046	0.127	0.029	0.142	0.9927	0.924	0.994	0.916
Median	0.027	0.126	0.011	0.138	1	1	1	1
Q_1	0.009	0.093	0.007	0.052	1	0.906	1	0.875
Q_3	0.061	0.16	0.023	0.212	1	1	1	1
IQR	0.052	0.067	0.016	0.159	0	0.094	0	0.125

Figure 18: Uncertainties over TP, FP, TN, and FN pixels for all 18 test scenes.

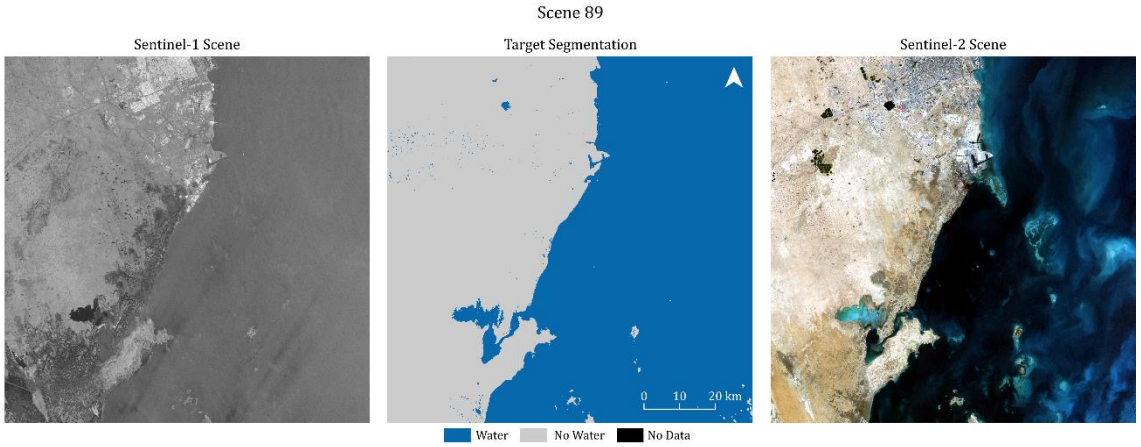
For UD1, higher variability is detected over correctly classified pixels. Figure 18 visualizes the distribution of the uncertainty values over TP, FP, TN, and FN pixels. It is a summary statistic for all 18 test scenes and all six predictions. This experiment does not provide an overview of the distribution at the scene level. For this reason, two scenes are also selected. The scenes and one zoomed sub-area within these scenes are visually assessed. Four classes are calculated to provide a better overview of the spatial distribution of the highly uncertain areas. They represent pixels that are AC, AU, IC, and IU. Pixels are labeled uncertain if the uncertainty exceeds 0.9 for UD1 or falls below 0.95 for UD2. Figures 19 and 20 provide an overview of test scenes 75 and 89 used as examples. The map provides the Sentinel-1 data, the generated ground truth mask, and the corresponding Sentinel-2 data. The scenes were chosen as examples as both demonstrate different geographical conditions as visible in the Sentinel-2 imagery. A table, as well as the corresponding overview maps containing information about all 18 test scenes, can be found in Appendix A.

Scene 75 is located in Russia close to the Ukrainian border and displays a coastline at the Sea of Azov. The coast region is mainly covered by agriculture. Scene 89 is located in Qatar. It depicts an area south of the capital Doha and lies at the Persian Gulf. It is a coastal area; the western part is mainly covered by the sea, while the western part consists of sand dunes and the coastline. The geographical conditions and their impact on the predictions of the BCNNs are further discussed in Chapter 7. Figures 21 – 24 show the maps products generated to visualize the prediction results and the spatial distribution of the uncertainty values. Figures 25 and 26 show a zoomed-in detail of scene 89. This is done to provide further information about the uncertainties. The map products for all scenes can be found in Appendix B. It also includes a more detailed map for Scene 31.



Scene ID	Valid Pixels	Main Landcover	Acquisition Date
75	188,259,427	Cultivated Vegetation	11-09-2019

Figure 19: Overview of test scene 75



Scene ID	Valid Pixels	Main Landcover	Acquisition Date
89	137,534,256	Sea	10-04-2020

Figure 20: Overview of test scene 89

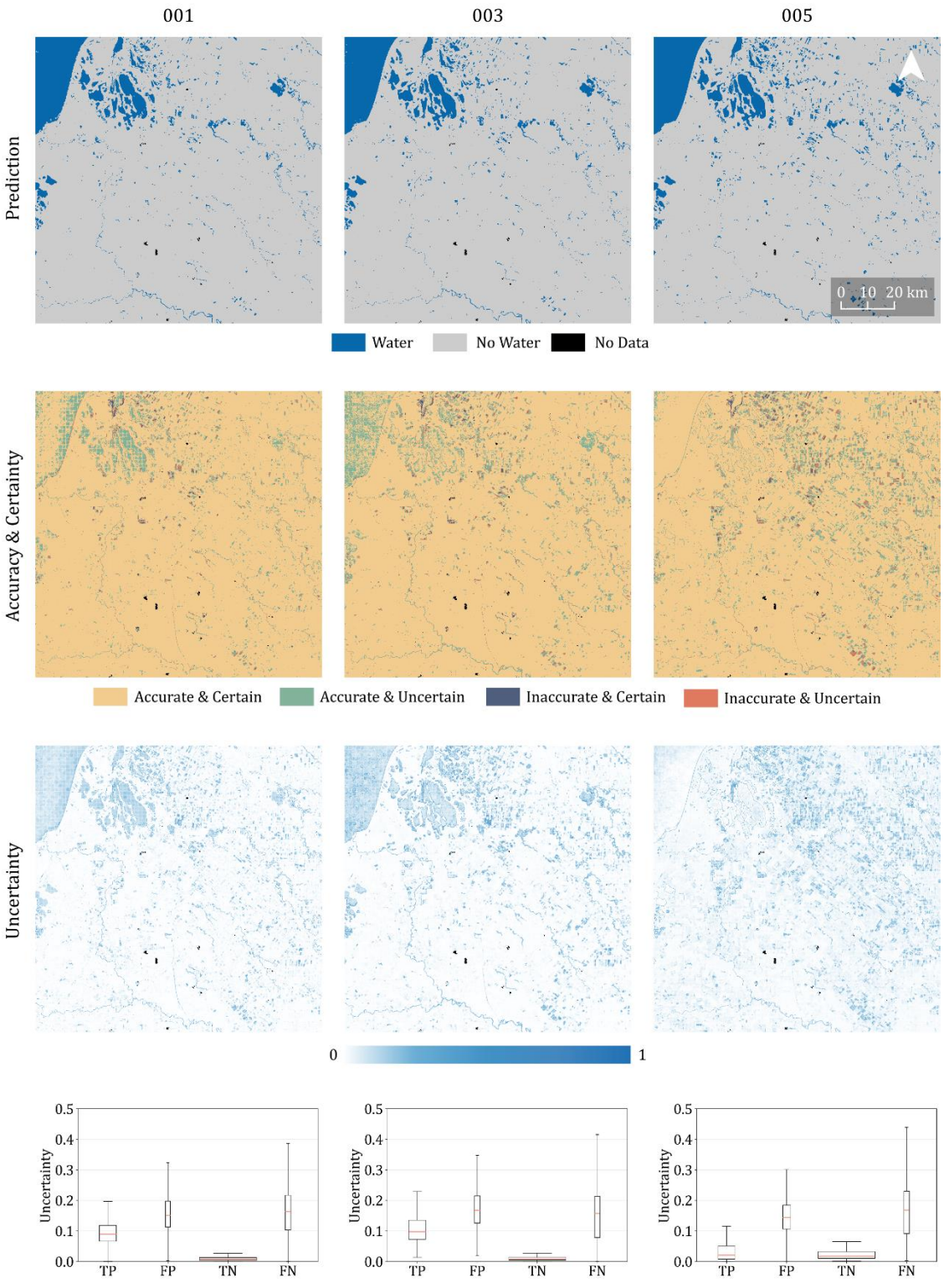


Figure 21: Visualization of highly uncertain areas for Scene 75 and UD1.: Prediction results (top), accuracy and certainty information (second row), uncertainty values (third row), and boxplots of uncertainties over TP, FP, TN, and FN (bottom).

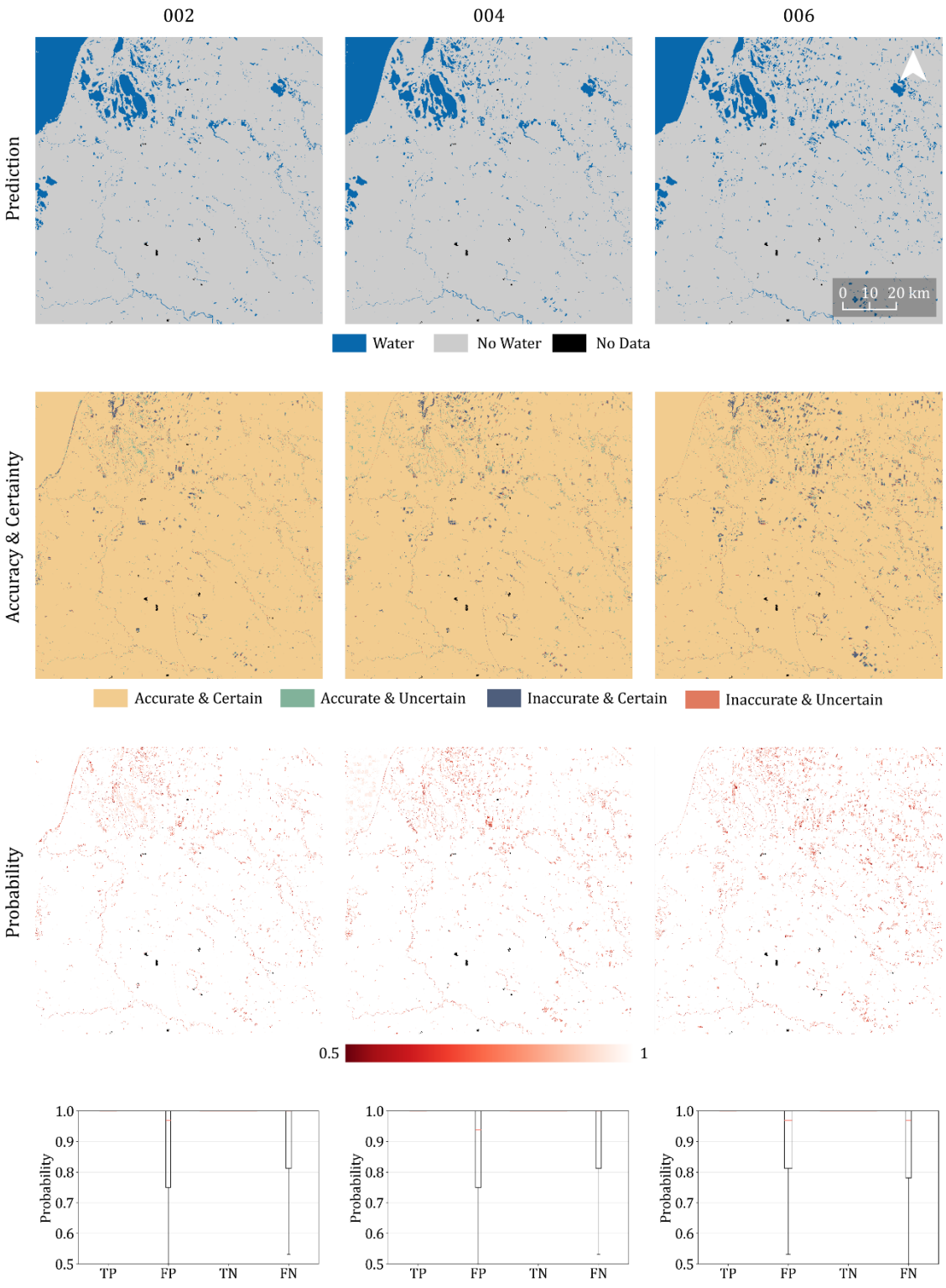


Figure 22: Visualization of highly uncertain areas for Scene 75 and UD2.: Prediction results (top), accuracy and certainty information (second row), probability values (third row), and boxplots of probabilities over TP, FP, TN, and FN (bottom).

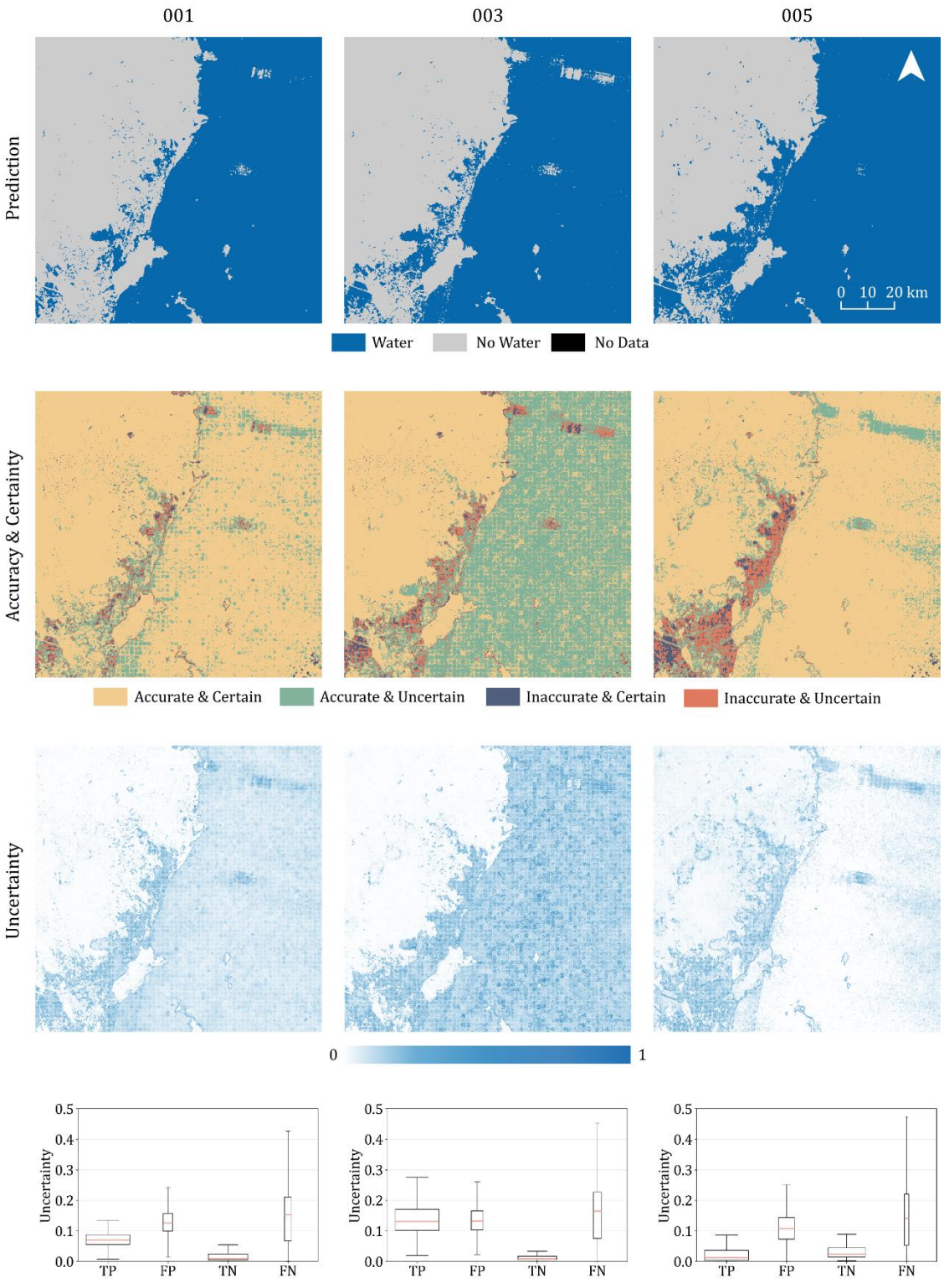


Figure 23: Visualization of highly uncertain areas for Scene 89 and UD1.: Prediction results (top), accuracy and certainty information (second row), uncertainty values (third row), and boxplots of uncertainties over TP, FP, TN, and FN (bottom).

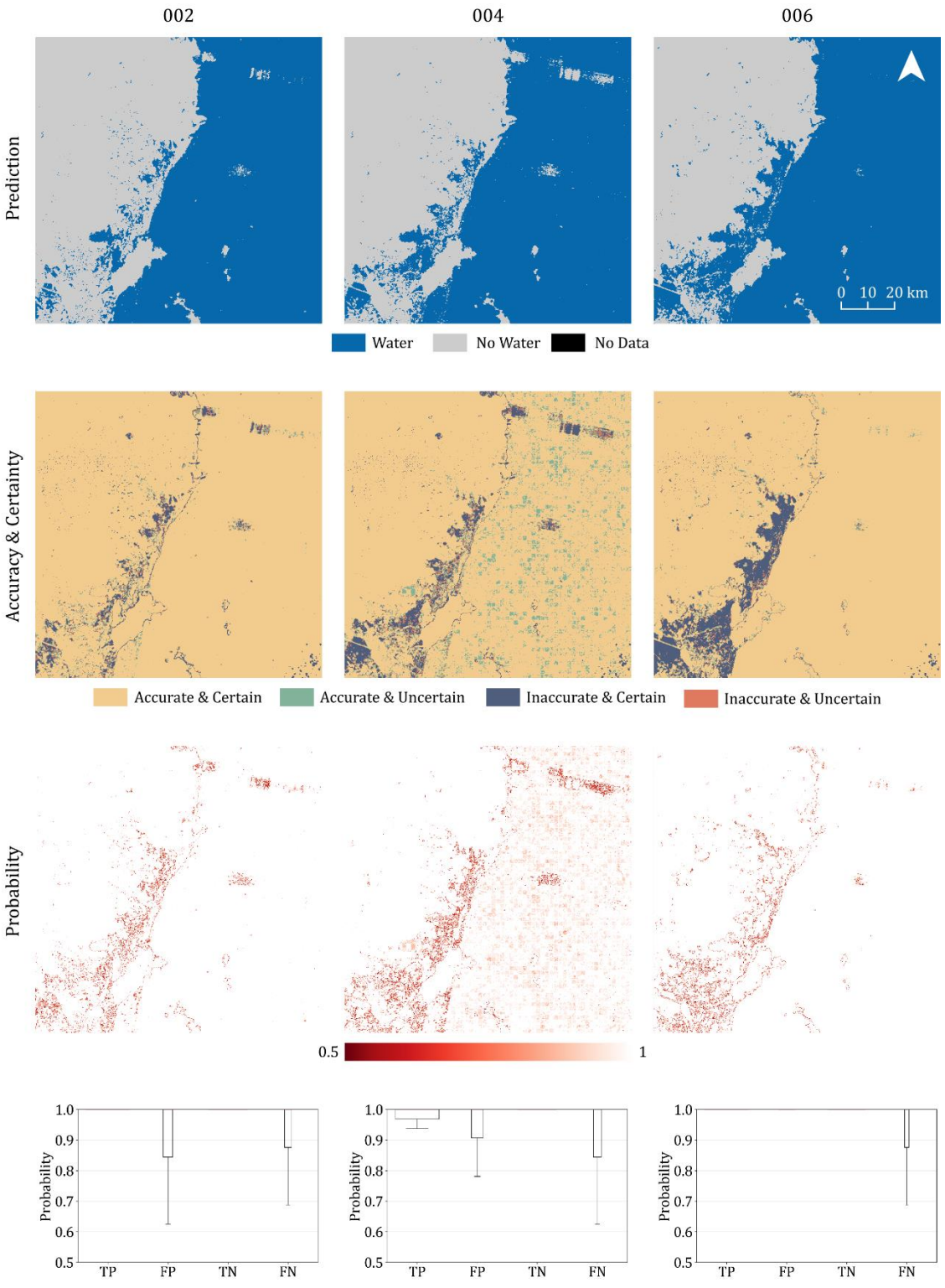


Figure 24: Visualization of highly uncertain areas for Scene 89 and UD2.: Prediction results (top), accuracy and certainty information (second row), probability values (third row), and boxplots of probabilities over TP, FP, TN, and FN (bottom).

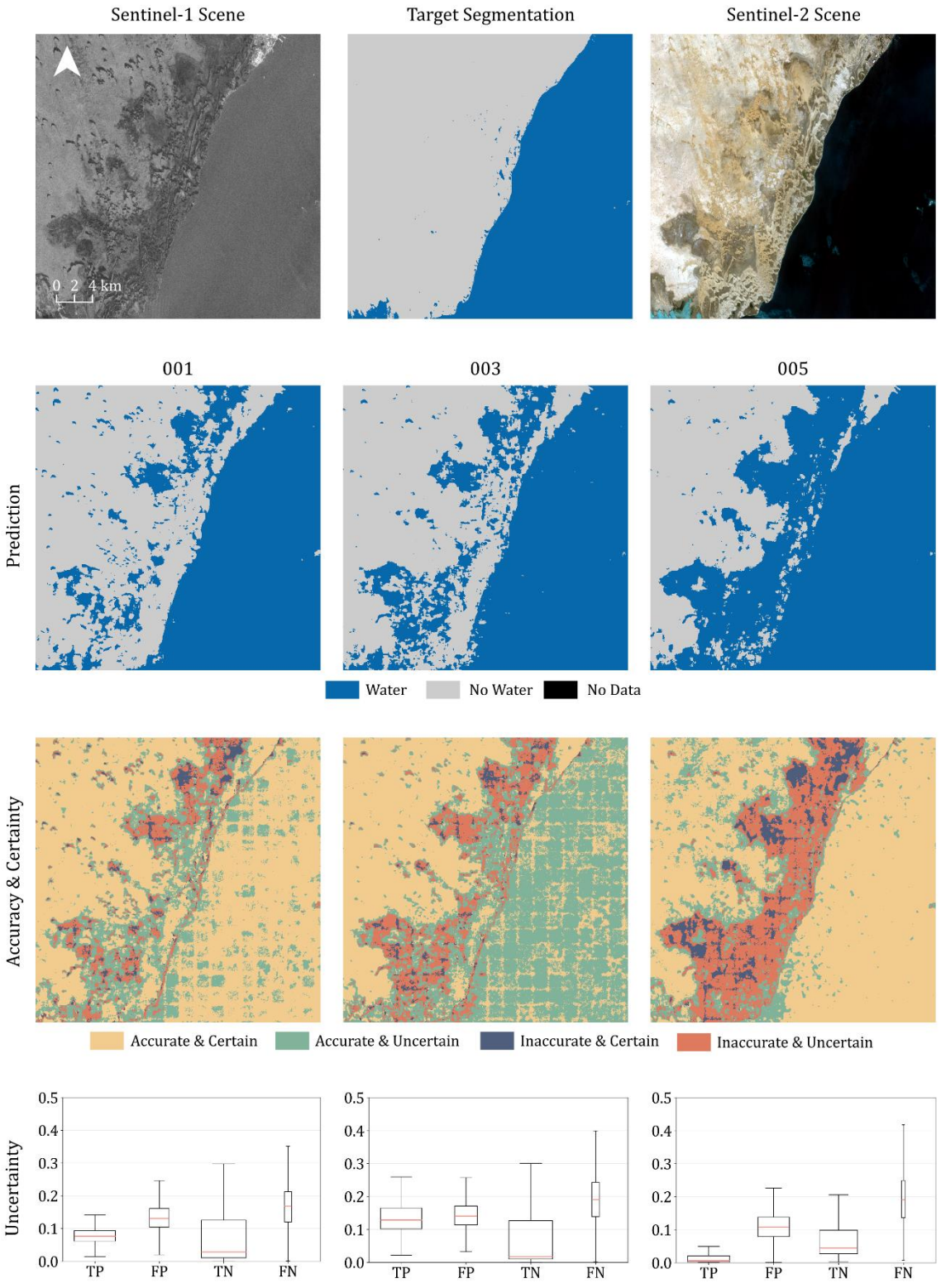


Figure 25: Visualization of highly uncertain areas for a zoomed-in detail in Scene 89 and for UD1.: Sentinel-1 data, target segmentation and Sentinel-data (top), Prediction results (second row), accuracy and certainty information (third row), and boxplots of uncertainties over TP, FP, TN, and FN (bottom).

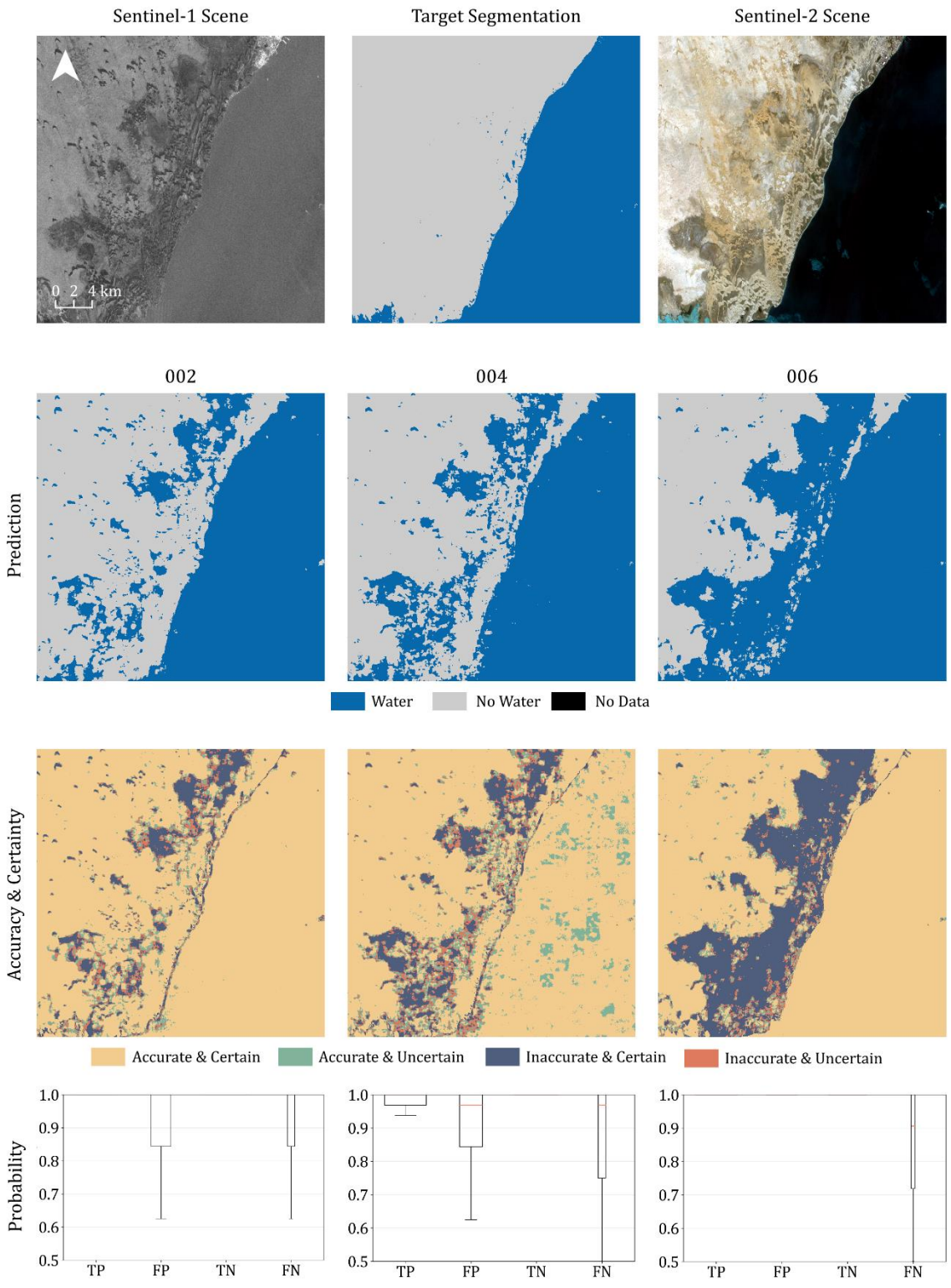


Figure 26: Visualization of highly uncertain areas for a zoomed-in detail in Scene 89 and for UD2.: Sentinel-1 data, target segmentation and Sentinel-data (top), Prediction results (second row), accuracy and certainty information (third row), and boxplots of uncertainties over TP, FP, TN, and FN (bottom).

In Figure 21, higher uncertainties in water areas of the prediction and the ground truth of Scene 75 can be detected for predictions 001 and 003. This also becomes evident when looking at the corresponding boxplots, as there is an elevation in the uncertainties over TP pixels. In contrast, the uncertainties for prediction 005 appear close to the boundary between water and non-water and in wrongly labeled areas. This can also be seen on the boxplot as there is a difference between the uncertainty levels of correctly labeled and misclassified pixels, the latter being higher than the former. For UD2 in Figure 22, the uncertainty appears to be higher only in the aforementioned border region. There is very little widespread uncertainty detectable compared to UD1.

Similar results can be found for Scene 89 in Figures 22 and 23. The uncertainties are elevated over open water areas for predictions 001 and 003 and appear higher over misclassified pixels in prediction 005. This, again, is also detectable in the matching boxplots. In Figure 23, the uncertainty seems to be elevated in the boundary regions of the water areas. The detailed view in Figures 24 and 25 magnifies those findings. Prediction 005 appears to provide the highest uncertainties over misclassified pixels and has the sharpest contrast of the uncertainties between correctly and wrongly labeled pixels.

6.3.2 Uncertainty values over different landcover classes

This section presents the analysis results regarding the distribution of uncertainties over different landcover types. This experiment was conducted by creating boxplots of the uncertainty values over the 12 landcover classes of the GCLS-LC100 dataset. The analysis was applied over all 18 test scenes and for all six predictions. Figure 27 displays the boxplots for the uncertainty values derived by UD1. The boxplots for UD2 are not shown in this section as the boxes, and the whiskers all lie close to a probability of 1 and are thus not visible in the plots.

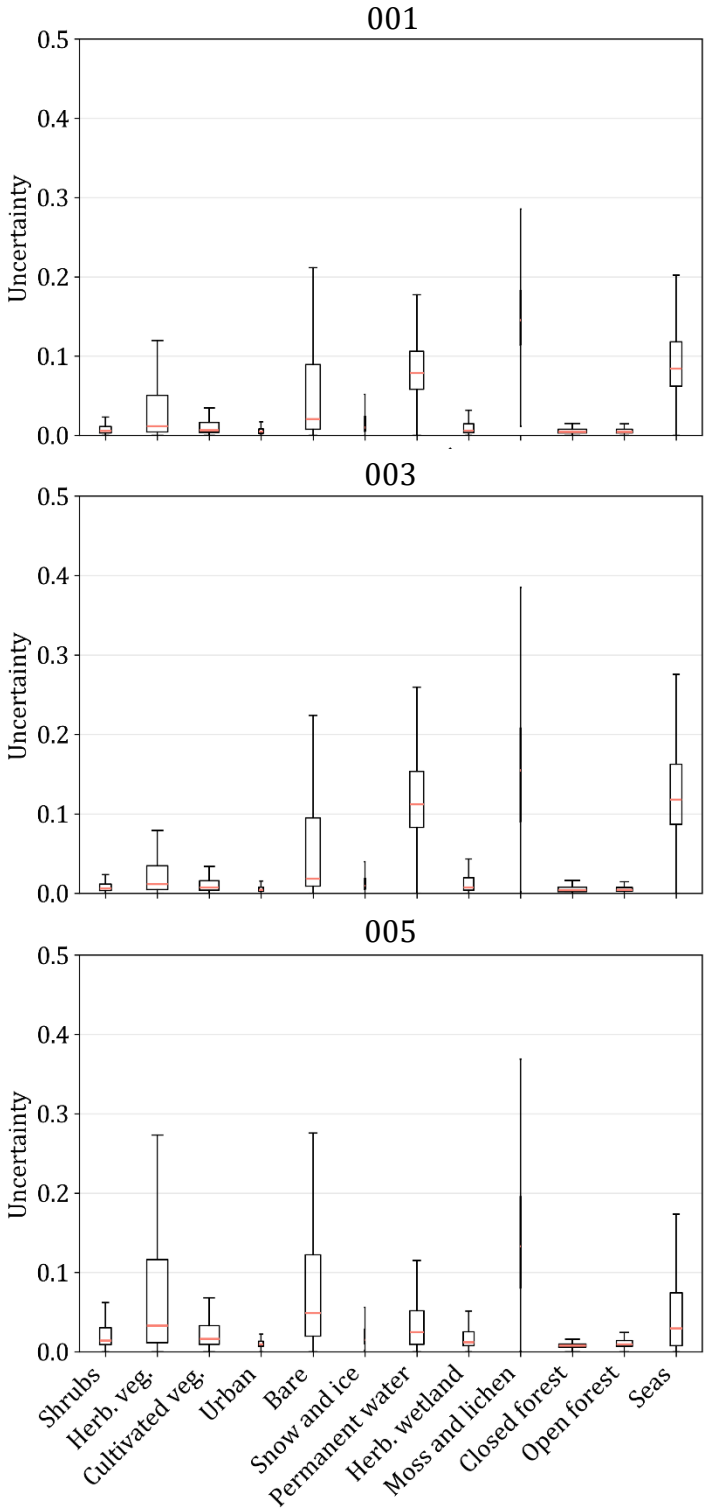


Figure 27: Distribution of the uncertainties for UD1 over the 12 landcover classes.

In all three boxplots, elevated uncertainty values can be detected over the classes Herbaceous Vegetation, Bare Soil, Permanent Water, Moss and Lichen, and Seas. The boxplots for predictions 001 and 003 show similar results. For prediction 005, differences from the other predictions are recognizable. There are higher uncertainties over the Herbaceous Vegetation class and lower uncertainties over the water classes Permanent Water and Seas. Despite its higher uncertainty values, it is also noted that the Moss and Lichen class is strongly underrepresented in the data. For UD2, the results were similar to what is presented in Figure 27. For predictions 002 and 004, there were higher uncertainties in Bare Soil, Herbaceous Wetland, and the water classes. For prediction 006, there were also higher uncertainties in the class Herbaceous Vegetation detectable.

6.4 Result optimization based on uncertainties

The final experiment in this study regards optimizing the prediction results based on the estimated uncertainty. On the basis of the findings in this study and the recommendations of Redekop & Chernyavskiy (2021), the experiments were only conducted for the prediction with uncertainty estimations for UD1. The reasoning is further discussed in section 7.4. A particular focus lies on the analysis of Prediction 005, which used model C, which was trained with the simplified data for one epoch training. Prediction 005 was chosen by visually assessing the boxplots in Figure 18. The difference between the means of the uncertainties for correctly and falsely classified pixels appears to be the highest. This means the uncertainties are higher over misclassified areas than the other predictions. The experiment also directly compared other predictions to confirm the choice. Figure 28 illustrates the progression of performance metrics based on the threshold above which the pixels were relabeled. The threshold was tested in 0.01 steps, and the metrics were calculated after each relabeling step. The analysis was done for all 18 test scenes and displayed the result for the whole model without applying morphological filtering. It is visible that the results worsen first for all predictions until they reach their initial prediction results. Only the Recall improves slightly for the predictions 003 and 005.

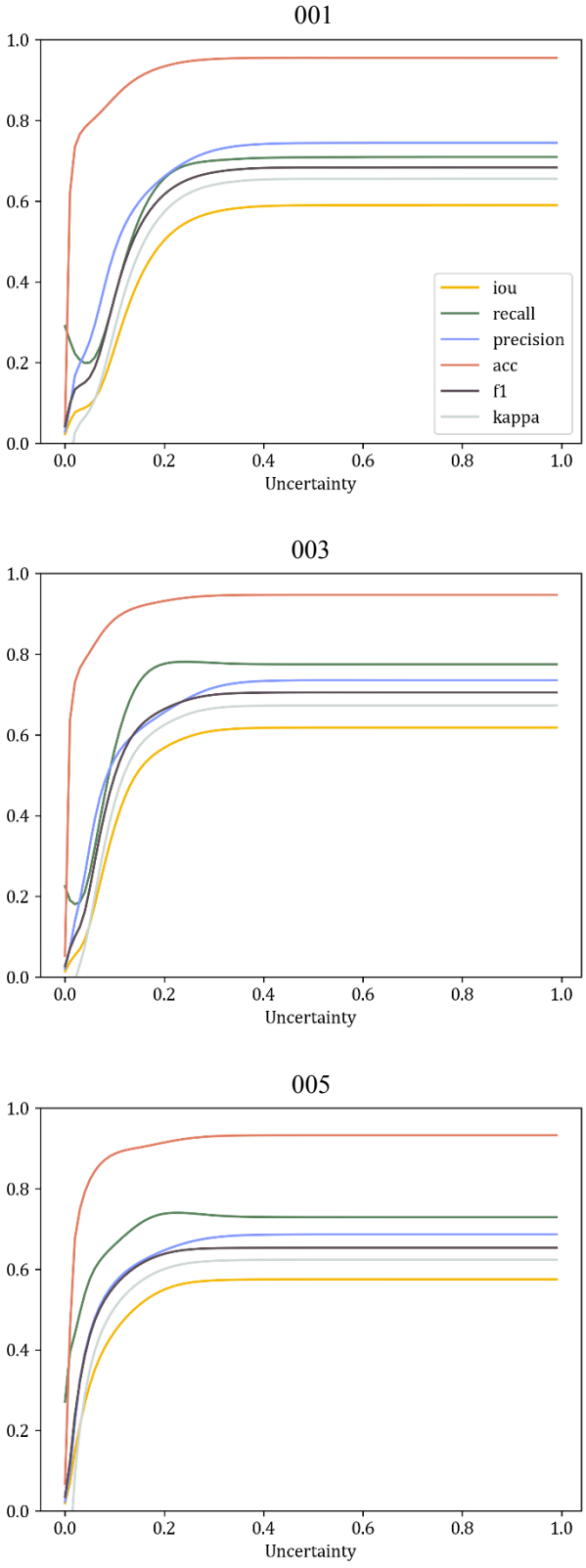


Figure 28: Performance metrics changes based on the chosen threshold for relabeling (0.01 steps) for Predictions 001, 003, and 005.

Figure 29 shows the results for Prediction 005 as the difference between the performance metrics after the relabeling and their initial performance. In addition, this performance change is illustrated after an erosion filter is applied to the relabeling mask. The used filter is a square structuring element with a size of 50 x 50 pixels.

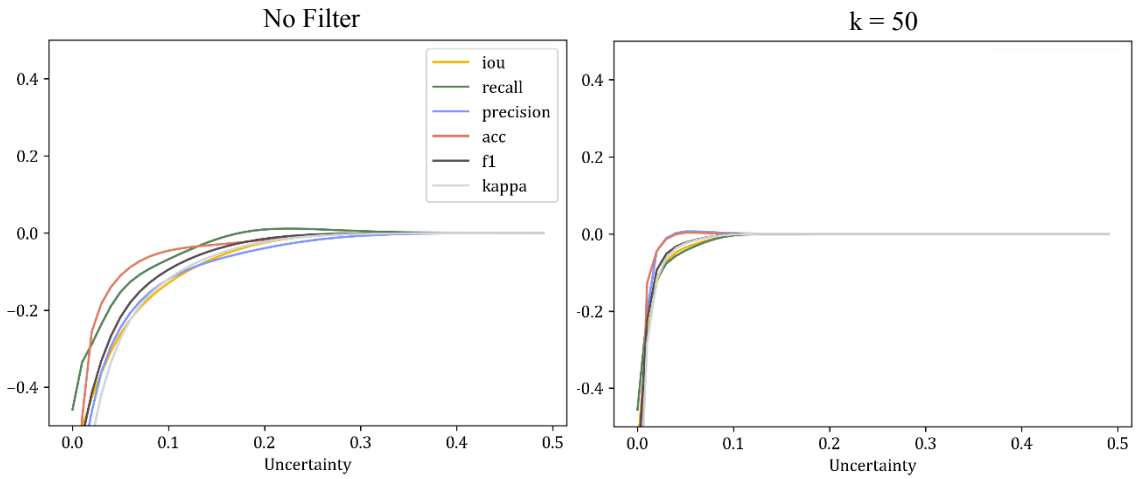


Figure 29: Change in performance metrics compared to the initial performance after relabeling for all 18 scenes and Prediction 005. No filter applied (left) and a square kernel (k=50) erosion (right).

Figures 30 and 31 provide further information on how different kernel sizes impact the relabeling process on a scene and on a more detailed basis. Figure 30 shows the effect of three different erosion filters applied to the predictions of scenes 75 and 89 for Prediction 005. Figure 31 illustrates the experiment in the detail area in scene 89. Here four different kernel sizes are compared. The graphs without an erosion filtering of the relabeling mask are also provided for comparison. The most considerable performance improvement can be achieved at an uncertainty threshold of 0.04 and 0.05 and a kernel size of 50 pixels for the square structuring element. Figure 32 shows exemplarily how the relabeling appears in a map product. The prediction water mask, the ground truth mask, and the masks after the relabeling process are provided. The figures illustrate the effect of the erosion filtering applied to the relabeling mask.

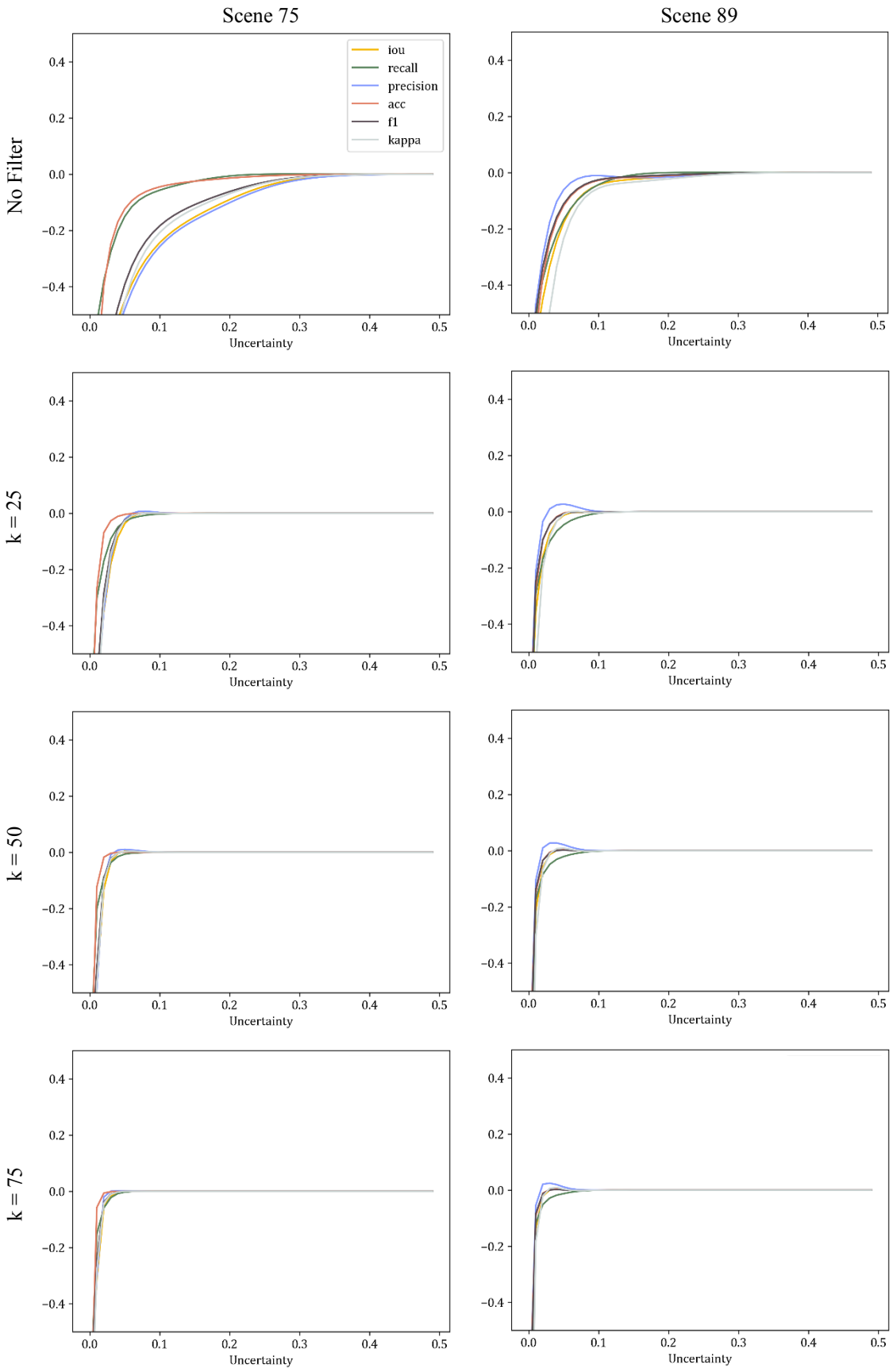


Figure 30: Relabeling for Scene 75 (left) and 89 (right). Change in performance metrics compared to the initial performance for prediction 005. No filter applied (top) and a square kernel erosion with dimensions: 25 x 25 (second row), 50 x 50 (third row) and 75 x 75 (bottom).

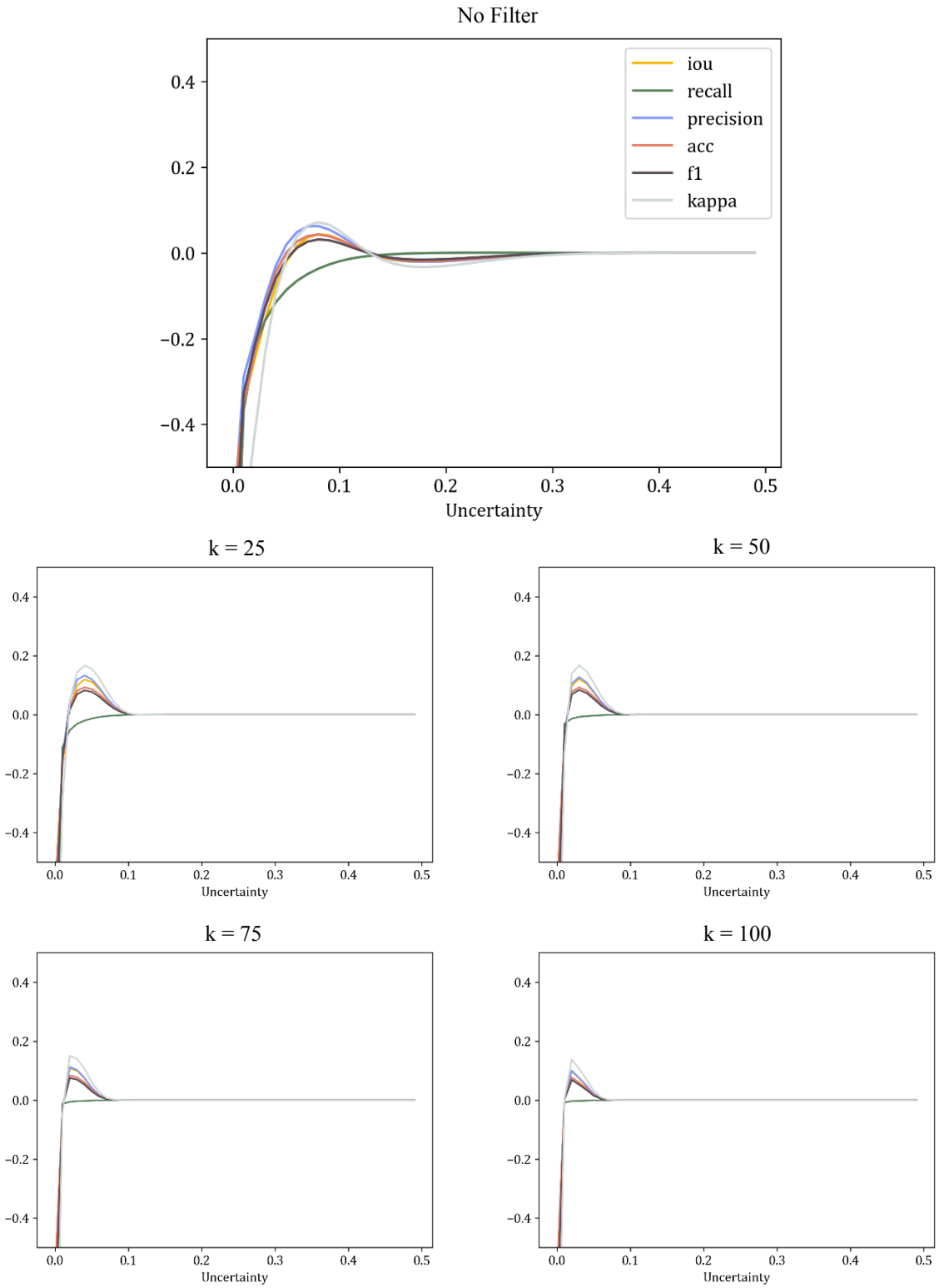


Figure 31: Relabeling for the zoomed-in detail of Scene 89. Change in performance metrics compared to the initial performance for prediction 005. No filter applied (top) and a square square kernel erosion with dimensions: 25 x 25 (middle left), 50 x 50 (middle right), 75 x 75 (bottom left) and 100 x 100 (bottom right).

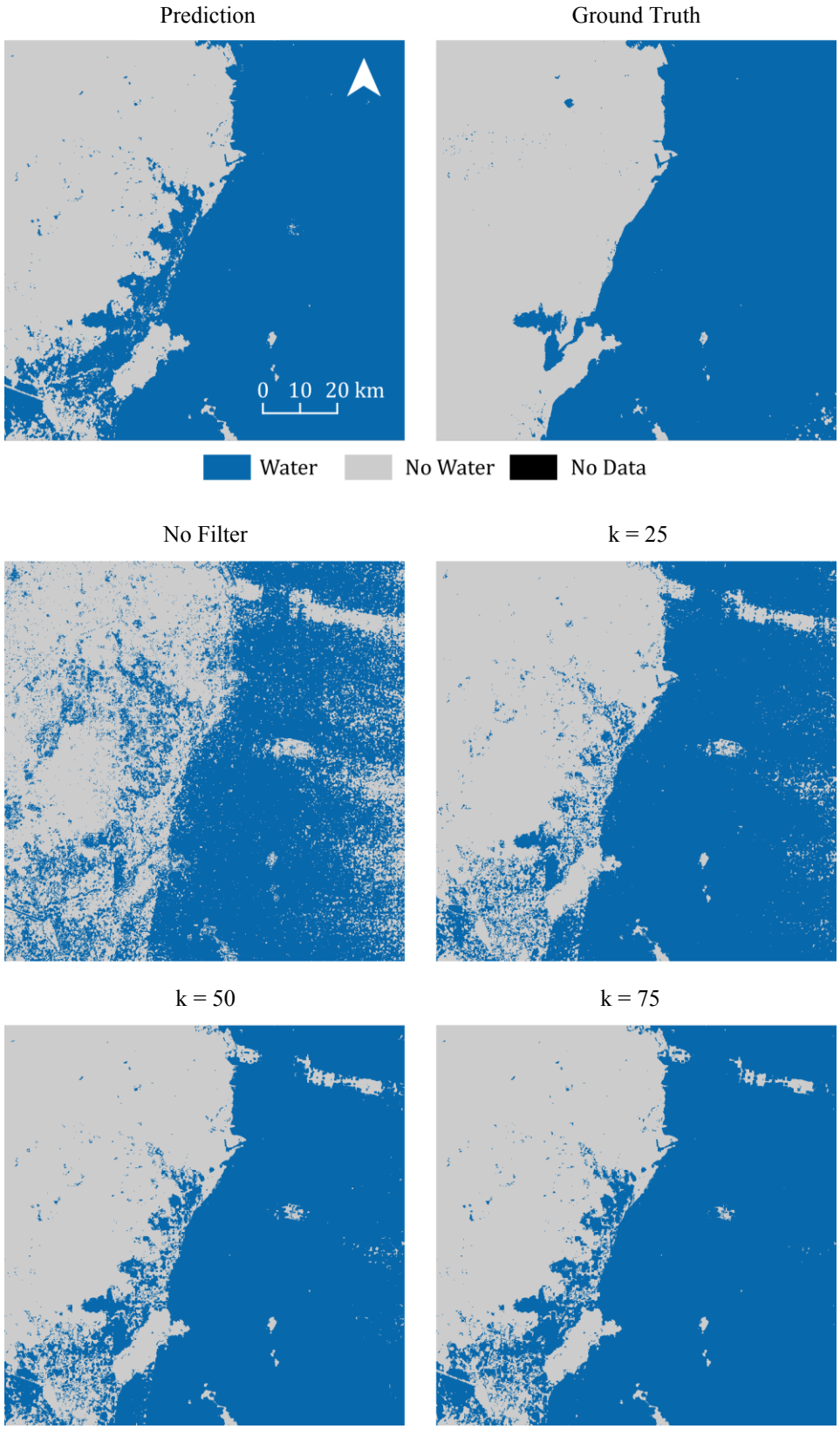


Figure 32: Maps of the conducted relabeling for scene 89 and a threshold of 0.05. Prediction (top left) and ground truth (top right) compared to relabeled masks without morphological filtering (middle left) and applied erosion to the relabeling mask with a square structuring element with kernel size: 25 x 25 (middle right), 50 x 50 (bottom left) and 75 x 75 (bottom right).

7 Discussion

Floodings are a relevant geographic topic in natural hazards and threaten socioeconomic and ecological structures and conditions. Deep Learning methods and SAR data are being utilized to minimize the adverse effects of flood events and improve existing management systems and flood mapping products. This study conducted experiments to determine how specific methods can improve flood mapping further. In detail, an analysis of the uncertainty estimations generated by a BCNN was performed. The results of the conducted study, presented in Chapter 6, are discussed in the following chapter. First, the model performance and the prediction result are mentioned (7.1), as well as the results of detecting the error-prone areas of those predictions (7.2). Then, the uncertainty analysis (7.3) and the relabeling of the predictions based on the uncertainties (7.4) are reviewed.

7.1 BCNN performance

Three different BCNNs were trained. The differences in the models can be found in the used training data and the model setups. Model A, trained with the best available training data for 12 epochs, produced good results that align with the findings of Hertel (2022). The other two models performed stepwise worse for most performance metrics, as seen in Table 6 and Figure 16. Model B was trained with the simplified data for 13 epochs, and model C with the simplified data for just one epoch. Model C performed the worst across all performance metrics except for the Recall, where model B performed slightly worse. Especially for the three metrics, F1-Score, the IoU, and the Kappa coefficient the stepwise decline is detectable.

The weaker performance also becomes evident in the training and validation losses and accuracies. By training only one epoch, the losses are higher compared to the final losses of models A and B. This also confirms the presumptions that worse training data produces worse results than the optimal data and fewer training epochs decrease the performance even further. This also matches the findings of Redekop & Chernyavsjiy (2021). The models produced uncertainty estimations based on both uncertainty definitions. The predictions were made for all 18 test scenes and all models, resulting in six predictions. There were only minimal differences in the performance metrics between the two predictions

for each model. The 32 individual predictions for one model prediction yielded different results. However, the ensemble mean of the prediction generated very similar results for the same model and input. This can lead to the conclusion that the model produces consistent results and offers the possibility to generate uncertainty estimations. This further proves the functionality of the approach by Hertel (2022).

7.2 Error-prone areas of the BCNN

The first experiment conducted with the predictions of the BCNNs aims to answer the question of what existing conditions occur in error-prone regions. This is tested over 12 different landcover classes acquired from the GCLS-LC100 dataset. The highest proportions of misclassified pixels were found over Bare Soil, Herbaceous Vegetation, and the water classes. The misclassifications over bare soil also match the presumption that smooth, natural surfaces cause overclassification of the water surface areas. Smooth surfaces reflect more significant portions of the emitted SAR radiation, thus possessing similar backscatter values as water surfaces (Martinis, Kuenzer, & Twele, 2015; Twele et al., 2016). Figure 17 also shows that the backscatter values are lower over bare soil, which further confirms the findings. The class Herbaceous Vegetation also poses lower backscatter values over the 18 test scenes. This fact might be the reason for overclassification in those areas. In future research, the classes Bare Soil and Herbaceous Vegetation should be further separated to gain more detailed insights into variabilities inside those classes. For example, Bare Soil merges areas like sand dunes and sparsely vegetated areas in one class. There might be significant differences in the error proneness of those subcategories (Buchhorn et al., 2020).

Datasets describing other conditions that can accompany misclassifications can also be added to the findings of this study. Those datasets can describe radar shadow phenomena or geomorphological conditions like slope or curvature. The slightly elevated proportion of misclassified pixels in the classes Permanent Water Body and Seas might be due to disturbances of the water detection in the boundary region between water and non-water. Here partially or entirely submerged vegetation can lead to misclassifications by the effects described in more detail in section 2.1.2 (Martinis, Kuenzer, & Twele, 2015; Shen et al., 2019). It should also be noted that in this study, one remote sensing product is compared to another. Both products are prone to different error sources and only reach

certain accuracies. The landcover data used in this thesis achieved an overall accuracy of 80.2%. However, the accuracy varies for the different landcover classes (Buchhorn et al., 2020). It is also noted that the used landcover data is an annually generated product that is prone to temporal decorrelation. Temporal decorrelation is shown as landcover can change over the course of a year (Lavalle, Simard and Hensley, 2011) . Therefore, the data might not match the conditions in the Sentinel-1 data. The results of this study are always to be assessed with these facts in mind. The landcover data also has a spatial resolution of 100 m compared to the 10 m of the Sentinel-1 data. This difference can also lead to inaccuracies in the analysis. In future research, it is advised to use data as close as possible to the input resolution. The findings of this study confirm the assumptions that bare soil has decreased backscatter values and a large proportion of the misclassified pixels lie within this class for all six predictions. It also shows that a high proportion of misclassified pixels lie within the Herbaceous Vegetation class, over which lower backscatter values were also detected.

7.3 Uncertainty analysis

The findings of the uncertainty analysis of whether misclassified pixels possess higher uncertainties compared to correctly labeled ones and whether there are landcover classes prone to higher uncertainties were presented in section 6.3. The analysis is also conducted to provide a knowledge basis for the experiments regarding the relabeling of pixels based on uncertainty. It is assumed that uncertainties are higher over misclassified pixels which are then relabeled. Figure 18 presents the uncertainty distribution results over TP, FP, TN, and FN pixels. For all six predictions, there are higher uncertainty values over misclassified pixels than over correctly labeled pixels. The most considerable difference in this regard is detectable for prediction 005. Here the mean and the median are substantially higher over FP and FN pixels. Those findings were also explored further in the examination of example scenes. Scene 75 consists mainly of agricultural areas and the Sea of Azov in southeastern Russia. The misclassified areas for the predictions appear to be over already harvested fields when visually comparing the areas with the Sentinel-2 data. In these areas, the uncertainty also appears higher, as shown in Figure 21. This seems to be especially true for prediction 005. For predictions 001 and 003, higher uncertainties can be detected over correctly labeled water areas. For the predictions estimated with UD2

(Figure 22), higher uncertainties can be found in the boundary regions between water and non-water. This finding also corresponds to the conclusion of Hertel (2022), as the uncertainties for UD2 are high in unclear pixels. That is not the case for pixels where the distribution of sigmoid values is either shifted towards water or non-water. As a result, the uncertainty estimations generated by UD1 are more spatially spread than those of UD2. The findings are also confirmed when visually assessing scene 89 (Figure 23-26).

The scene displays an arid area in Qatar close to the coast that includes water areas of the Persian Gulf. The coastal area consists mainly of dry, sandy plains (Engel et al., 2018). The misclassifications in this area most probably occur because of the smooth sand surfaces that can lead to overclassification (Martinis, Kuenzer, & Twele, 2015). The uncertainties of UD1 have a more spread-out distribution compared to UD2. When further assessing the uncertainties estimated by UD1, it is visible that for predictions 001 and 003, the uncertainties are elevated over correctly labeled water areas. For prediction 005, the highly uncertain areas resemble the misclassified ones better. This is visible in scenes 75 and 89. It is also recognized that IU areas are surrounded by IC pixels. This may also be due to the chosen threshold above which pixels are labeled as uncertain. The maps displaying the uncertainties for UD1 show horizontal and vertical patterns. This can be caused by the decrease of uncertainties towards the edges of the predicted image, as the predictions are conducted on a tile basis (Hertel, 2022). The BCNN adapted by Hertel (2022) for SAR-based image segmentation also includes an additional parameter to define the overlap of the tiles used for the prediction. The horizontal and vertical phenomena might decrease when this value is set higher to include more pixels surrounding each tile and thus, increasing the overlap.

The second part of the uncertainty analysis regards the uncertainties' distribution over different landcover types. Higher uncertainties can be found over different landcover types, as described in section 6.3.1. The interpretation of the boxplots and the descriptive statistics lead to the conclusion that there are elevated uncertainties over the classes Herbaceous Vegetation, Bare Soil, Permanent Water, Moss and Lichen, and Seas. This proves especially for models A and B. Model C shows higher uncertainties over the class Herbaceous Vegetation and lower values over the water classes. When assessing the results visually, there is a similarity between the boxplots for prediction 005 and the backscatter values over the landcover classes. Therefore, the classes possessing lower backscatter values resemble higher uncertainties. This might be due to the higher

misclassifications over those classes and, thus, higher corresponding uncertainty values. This concerns the findings of the uncertainty analysis over misclassified areas. This analysis could also be extended with additional datasets, as only landcover classes are considered in this study.

7.4 Applicability of relabeling based on uncertainties

Based on the findings discussed in 7.3, the uncertainties of UD1 and, more precisely, prediction 005 were found to be the most suitable for utilization in the relabeling process. Therefore, most experiments were conducted for prediction 005 of model C. For this prediction, the uncertainties are higher over misclassified areas than in the other predictions. Model C was trained with the simplified data. This data only shows one level of geometric simplification. In future research, the impact of varying levels of introduced noise can be analyzed to test their impact on the results further. Here, different error sources in human and automatically labeling might be reproduced. In this study, the relabeling of all test scenes only led to slight increases in the Recall metric. Compared to predictions 001 and 003, the increase in performance was the highest for prediction 005 as there was a slight increase in the Recall and F1-Score detectable. This seems to confirm the findings of Redekop & Chernyavkiy (2021) that one epoch training is more suitable for the relabeling process compared to more epochs. Concerning the erosion filtering of the relabeling mask, the study shows an improvement in the performance metrics when applying the filter to the relabeling mask before relabeling. This can be detected for all scenes and also on a more detailed level.

Furthermore, the influence of different sizes of a square filter kernel was tested. After assessing the results quantitatively and qualitatively, it can be assumed that a square filter of dimensions 50 x 50 pixels produces the most remarkable performance improvements. This accounts for the produced result but may vary when applied to different areas. The chosen uncertainty threshold, above which the pixels are relabeled, also needs to be considered. According to the findings in this study, when applying a 50 x 50 erosion filter, the optimal threshold is 0.05, resulting in the highest increase in the performance metrics. It is noted that the optimal threshold varies strongly based on the chosen filter kernel size, as visible in Figure 31. This research could be extended to assess further the impact of different morphological filtering before and after the relabeling and their consequences

for the optimal uncertainty threshold for relabeling. This experiment proved the applicability of the proposed method for certain areas. The method might reduce the resources needed to create training data for flood mapping applications.

8 Conclusions

This chapter summarizes the results presented in this study. It also provides recommendations for future research building on the results.

8.1 Summary

Flood mapping poses a relevant tool to support flood management during and after a flooding event. It is, therefore, necessary to provide accurate and timely data. BCNNs have produced highly accurate classification results for water detections in SAR data. The approach by Hertel (2022) was proven and further tested in this study. Three BCNNs were trained using different setups and training data. The training data consists of Sentinel-1 data and the corresponding reference masks generated from Sentinel-2 data and by expert knowledge. In addition, another training dataset was created using a line simplification algorithm to simplify the geometries of the data. As expected, the model trained with the best available training data achieved the best prediction results. The second model was trained with the simplified training data for 13 epochs and achieved lower average performance metrics. The third model was trained with the simplified data for only one epoch and achieved the weakest performance among the three models for the 18 test scenes.

Regarding error-prone areas, the landcover class Bare Soil achieved high levels of misclassified pixels. This is probably due to the smooth surface of sandy and comparable areas.

For the uncertainty analysis, the study has confirmed that the uncertainty values are higher over misclassified pixels than correctly classified pixels. This appears to be especially visible for the model trained with the simplified data and for one epoch (Model C). The difference in the mean values is higher for the uncertainty definition calculated by the spread of sigmoid values (UD1). It is therefore concluded that model C and UD1 are best suited for improving the results based on the uncertainty.

The uncertainty values appear elevated over certain classes like Bare Soil and Herbaceous Vegetation. This also matches the experiment's findings regarding the condition of erroneous pixels.

Based on the assumption that uncertainties are higher over misclassified pixels, the pixels of high uncertainty were relabeled to their opposite class. The study showed promising results. An improvement is detectable in certain areas. The most considerable improvements in the performance metrics were detected for model C and UD1. The results improved further when applying morphological erosion filtering to the mask of the highly uncertain pixels. According to the findings in this study, the best results were achieved when applying a 50 x 50 erosion filter with an uncertainty threshold of 0.05. This is determined for model C and UD1. It is concluded that the approach might be suitable to decrease the resources needed to create appropriate reference data for training. This could consequently lead to trained models with data that covers more geographically diverse areas. This might also result in more complete models that can be used globally.

8.2 Recommendations for Future Research

This study's research questions were mainly answered but still left some parts unanswered, and new questions have arisen. These might be subject to future research based on the findings in this thesis.

Additional datasets can be implemented to analyze the conditions of error-prone areas and the uncertainties. This could mean extending the experiments to conditions assumed to cause difficulties for the model, like radar shadows or submerged vegetation. For the landcover data, it might be reasonable to separate the classes Bare Soil and Herbaceous Vegetation further to gain more detailed insights on variabilities inside those classes. It is also recommended to use inter-annual data to avoid temporal decorrelation.

Regarding the relabeling based on uncertainty, different degrees of simplification of the training data should be tested. This can simulate different error sources that persist when creating the data. The difference between different creators of the training data could also be tested. Different creators can label the same data to highlight differences between different ways of creation. Future research might further emphasize the impact of different morphological filtering before and after the relabeling process.

Bibliography

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., … Zheng, X. (2016). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems*. https://doi.org/10.48550/arxiv.1603.04467

Abiodun, O. I., Jantan, A., Omolara, A. E., Dada, K. V., Mohamed, N. A., & Arshad, H. (2018). State-of-the-art in artificial neural network applications: A survey. *Heliyon*, *4*(11), e00938. https://doi.org/10.1016/j.heliyon.2018.e00938

Albawi, S., Bayat, O., Al-Azawi, S., & Ucan, O. N. (2018). Social Touch Gesture Recognition Using Convolutional Neural Network. *Computational Intelligence and Neuroscience*, *2018*, 1–10. https://doi.org/10.1155/2018/6973103

Albawi, S., Mohammed, T. A., & Al-Zawi, S. (2017). Understanding of a convolutional neural network. *2017 International Conference on Engineering and Technology (ICET)*, 1–6. https://doi.org/10.1109/ICEngTechnol.2017.8308186

Allan, E., Bossdorf, O., Dormann, C. F., Prati, D., Gossner, M. M., Tscharntke, T., Blüthgen, N., Bellach, M., Birkhofer, K., Boch, S., Böhm, S., Börschig, C., Chatzinotas, A., Christ, S., Daniel, R., Diekötter, T., Fischer, C., Friedl, T., Glaser, K., … Fischer, M. (2014). Interannual variation in land-use intensity enhances grassland multidiversity. *Proceedings of the National Academy of Sciences of the United States of America*, *111*(1), 308–313. https://doi.org/10.1073/PNAS.1312213111/SUPPL_FILE/PNAS.201312213SI.PDF

Anusha, N., & Bharathi, B. (2020). Flood detection and flood mapping using multi-temporal synthetic aperture radar and optical data. *The Egyptian Journal of Remote Sensing and Space Science*, *23*(2), 207–219. https://doi.org/10.1016/j.ejrs.2019.01.001

Bamler, R. (2000). Principles Of Synthetic Aperture Radar . *Surveys in Geophysics*, *21*(2/3), 147–157. https://doi.org/10.1023/A:1006790026612

Barredo, J. I. (2009). Normalised flood losses in Europe: 1970–2006. *Natural Hazards and Earth System Sciences*, *9*(1), 97–104. https://doi.org/10.5194/NHESS-9-97-2009

Bayes, T. (1763). LII. An essay towards solving a problem in the doctrine of chances. By the late Rev. Mr. Bayes, FRS communicated by Mr. Price, in a letter to John Canton, AMFR S. *Philosophical Transactions of the Royal Society of London* , *53*.

Bertram, A., Wendleder, A., Schmitt, A., & Huber, M. (2016). Long-Term Monitoring of Water Dynamics in the Sahel Region using the Multi-SAR-System. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *XLI-B8*, 313–320. https://doi.org/10.5194/isprs-archives-XLI-B8-313-2016

Blundell, C., Cornebise, J., Kavukcuoglu, K., & Wierstra, D. (2015). *Weight Uncertainty in Neural Networks*. http://arxiv.org/abs/1505.05424

Bokwa, A. (2013). *Encyclopedia of Natural Hazards* (P. T. Bobrowsky, Ed.). Springer Netherlands. https://doi.org/10.1007/978-1-4020-4399-4

Bolstad, W., & Curran, J. (2016). *Introduction to Bayesian Statistics* (3rd ed.). Wiley.

Bonafilia, D., Tellman, B., Anderson, T., & Issenberg, E. (2020). Sen1Floods11: a georeferenced dataset to train and test deep learning flood algorithms for Sentinel-1. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 835–845. https://doi.org/10.1109/CVPRW50498.2020.00113

Bossard, M., Feranec, J., & Otahel, J. (2000). *CORINE land cover technical guide: Addendum 2000* (Vol. 40). European Environmental Agency.

Brivio, P. A., Colombo, R., Maggi, M., & Tomasoni, R. (2002). Integration of remote sensing data and GIS for accurate mapping of flooded areas. *International Journal of Remote Sensing*, *23*(3), 429–441. https://doi.org/10.1080/01431160010014729

Bryant, E. (2005). *An introduction to natural hazards*. www.cambridge.org (*accessed 08-25-2022*)

Buchhorn, M., Bertels, L., Smets, B., Lesiv, M., & Tsendbazar, N.-E. (2019). *Copernicus Global Land Service: Land Cover 100m: version 2 Globe 2015: Algorithm Theoretical Basis Document*. https://doi.org/10.5281/ZENODO.3606446

Buchhorn, M., Lesiv, M., Tsendbazar, N.-E., Herold, M., Bertels, L., & Smets, B. (2020). Copernicus Global Land Cover Layers—Collection 2. *Remote Sensing*, *12*(6), 1044. https://doi.org/10.3390/rs12061044

Cohen, J. (1960). A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, *20*(1), 37–46. https://doi.org/10.1177/001316446002000104

Dera, D., Rasool, G., Bouaynaya, N. C., Eichen, A., Shanko, S., Cammerata, J., & Arnold, S. (2020). Bayes-SAR net: Robust SAR image classification with uncertainty estimation using Bayesian convolutional neural network. *2020 IEEE International Radar Conference, RADAR 2020*, 362–367. https://doi.org/10.1109/RADAR42522.2020.9114737

Deutsch, M., & Ruggles, F. (1974). Optical Data Processing und Projected Applications of the ERTS-1 Imagery Covering the 1973 Mississippi River Valley floods. *Journal of the American Water Resources Association*, *10*(5), 1023–1039. https://doi.org/10.1111/j.1752-1688.1974.tb00622.x

Dorst, L., & van den Boomgaard, R. (1994). Morphological signal processing and the slope transform. *Signal Processing*, *38*(1), 79–98. https://doi.org/10.1016/0165-1684(94)90058-2

Eberenz, J., Verbesselt, J., Herold, M., Tsendbazar, N.-E., Sabatino, G., & Rivolta, G. (2016). Evaluating the Potential of PROBA-V Satellite Image Time Series for Improving LC Classification in Semi-Arid African Landscapes. *Remote Sensing*, *8*(12), 987. https://doi.org/10.3390/rs8120987

EM-DAT. (2022). *The International Disaster Database, Centre for research on Epide- miology of Disasters.* www.emdat.be (*accessed 08-10-2022*)

Engel, M., Boesl, F., & Brückner, H. (2018). Migration of Barchan Dunes in Qatar–Controls of the Shamal, Teleconnections, Sea-Level Changes and Human Impact. *Geosciences 2018, Vol. 8, Page 240*, *8*(7), 240. https://doi.org/10.3390/GEOSCIENCES8070240

ESA. (2021). *Sentinel-1 Acquisition Modes*. https : //sentinel.esa.int/web/sentinel/user-guides/sentinel-1-sar/acquisition-modes (*accessed 08-14-2022*)

Foody, G. M. (2002). Status of land cover classification accuracy assessment. *Remote Sensing of Environment*, *80*(1), 185–201. https://doi.org/10.1016/S0034-4257(01)00295-4

Fritz, S., McCallum, I., Schill, C., Perger, C., See, L., Schepaschenko, D., van der Velde, M., Kraxner, F., & Obersteiner, M. (2012). Geo-Wiki: An online platform for improving global land cover. *Environmental Modelling and Software*, *31*, 110–123. https://doi.org/10.1016/j.envsoft.2011.11.015

Gal, Y., & Ghahramani, Z. (2015). *Bayesian Convolutional Neural Networks with Bernoulli Approximate Variational Inference*. http://arxiv.org/abs/1506.02158

Gevrey, M., Dimopoulos, I., & Lek, S. (2003). Review and comparison of methods to study the contribution of variables in artificial neural network models. *Ecological Modelling*, *160*(3), 249–264. https://doi.org/10.1016/S0304-3800(02)00257-0

Ghemawat, S., Zheng, X., Moore, S., Abadi, M., Dean, J., Chen, Z., Kudlur, M., Warden, P., Irving, G., Chen, J., Barham, P., Yu, Y., Vasudevan, V., Devin, M., Tucker, P., Davis, A., Steiner, B., Monga, R., Wicke, M., … Isard, M. (2016). TensorFlow: A system for large-scale machine learning. *Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation, OSDI 2016*, 265–283.

Gorelick, N., Hancher, M., Dixon, M., Ilyushchenko, S., Thau, D., & Moore, R. (2017). Google Earth Engine: Planetary-scale geospatial analysis for everyone. *Remote Sensing of Environment*, *202*, 18–27. https://doi.org/10.1016/j.rse.2017.06.031

Gstaiger, V., Huth, J., Gebhardt, S., Wehrmann, T., & Kuenzer, C. (2012). Multi-sensoral and automated derivation of inundated areas using TerraSAR-X and ENVISAT ASAR data. *International Journal of Remote Sensing*, *33*(22), 7291–7304. https://doi.org/10.1080/01431161.2012.700421

Gu, J., Wang, Z., Kuen, J., Ma, L., Shahroudy, A., Shuai, B., Liu, T., Wang, X., Wang, G., Cai, J., & Chen, T. (2018). Recent advances in convolutional neural networks. *Pattern Recognition*, *77*, 354–377. https://doi.org/10.1016/j.patcog.2017.10.013

Gutman, G., Huang, C., Chander, G., Noojipady, P., & Masek, J. G. (2013). Assessment of the NASA-USGS Global Land Survey (GLS) datasets. *Remote Sensing of Environment*, *134*, 249–265. https://doi.org/10.1016/j.rse.2013.02.026

Haut, J. M., Paoletti, M. E., Plaza, J., Li, J., & Plaza, A. (2018). Active Learning With Convolutional Neural Networks for Hyperspectral Image Classification Using a New Bayesian Approach. *IEEE Transactions on Geoscience and Remote Sensing*, *56*(11), 6440–6461. https://doi.org/10.1109/TGRS.2018.2838665

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778. https://doi.org/10.1109/CVPR.2016.90

Heijmans, H. J. A. M., & Ronse, C. (1990). The algebraic basis of mathematical morphology I. Dilations and erosions. *Computer Vision, Graphics and Image Processing*, *50*(3), 245–295. https://doi.org/10.1016/0734-189X(90)90148-O

Helleis, M., Wieland, M., Krullikowski, C., Martinis, S., & Plank, S. (2022). Sentinel-1-Based Water and Flood Mapping: Benchmarking Convolutional Neural Networks Against an Operational Rule-Based Processing Chain. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *15*, 2023–2036. https://doi.org/10.1109/JSTARS.2022.3152127

Heremans, R., Willekens, A., Borghys, D., Verbeeck, B., Valckenborgh, J., Acheroy, M., & Perneel, C. (2003). Automatic detection of flooded areas on ENVISAT/ASAR images using an object-oriented classification technique and an active contour algorithm. *International Conference on Recent Advances in Space Technologies, 2003. RAST '03. Proceedings Of*, 311–316. https://doi.org/10.1109/RAST.2003.1303926

Herrera-Cruz, V., & Koudogbo, F. (2009). TerraSAR-X Rapid mapping for flood events. *Proceedings of the International Society for Photogrammetry and Remote Sensing, Earth Imaging for Geospatial Information*.

Hertel, V. (2022). *Probabilistic Deep Learning Methods for Capturing Uncertainty in SAR-based Water Segmentation Maps*. DLR e.V.

Homer, C., Dewitz, J., Jin, S., Xian, G., Costello, C., Danielson, P., Gass, L., Funk, M., Wickham, J., Stehman, S., Auch, R., & Riitters, K. (2020). Conterminous United States land cover change patterns 2001–2016 from the 2016 National Land Cover Database. *ISPRS Journal of Photogrammetry and Remote Sensing*, *162*, 184–199. https://doi.org/10.1016/j.isprsjprs.2020.02.019

Hüllermeier, E., & Waegeman, W. (2021). Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. *Machine Learning*, *110*(3), 457–506. https://doi.org/10.1007/s10994-021-05946-3

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science and Engineering*, *9*(3), 90–95. https://doi.org/10.1109/MCSE.2007.55

Jacobs, R. A. (1988). Increased rates of convergence through learning rate adaptation. *Neural Networks*, *1*(4), 295–307. https://doi.org/10.1016/0893-6080(88)90003-2

Jain, S. K., Singh, R. D., Jain, M. K., & Lohani, A. K. (2005). Delineation of flood-prone areas using remote sensing techniques. *Water Resources Management*, *19*(4), 333–347. https://doi.org/10.1007/s11269-005-3281-5

Joshaghani, M., Davari, A., Hatamian, F. N., Maier, A., & Riess, C. (2022). *Bayesian Convolutional Neural Networks for Limited Data Hyperspectral Remote Sensing Image Classification*.

Jospin, L. V., Laga, H., Boussaid, F., Buntine, W., & Bennamoun, M. (2022). Hands-On Bayesian Neural Networks—A Tutorial for Deep Learning Users. *IEEE Computational Intelligence Magazine*, *17*(2), 29–48. https://doi.org/10.1109/MCI.2022.3155327

Kang, W., Xiang, Y., Wang, F., Wan, L., & You, H. (2018). Flood Detection in Gaofen-3 SAR Images via Fully Convolutional Networks. *Sensors*, *18*(9), 2915. https://doi.org/10.3390/s18092915

Keiron, O., & Nash, R. (2015). An introduction to convolutional neural networks. *ArXiv Preprint*, *arXiv:1511.08458*.

Kiureghian, A. der, & Ditlevsen, O. (2009). Aleatory or epistemic? Does it matter? *Structural Safety*, *31*(2), 105–112. https://doi.org/10.1016/j.strusafe.2008.06.020

Klemas, V. (2015). Remote sensing of floods and flood-prone areas: An overview. In *Journal of Coastal Research* (Vol. 31, Issue 4, pp. 1005–1013). Coastal Education Research Foundation Inc. https://doi.org/10.2112/JCOASTRES-D-14-00160.1

Köhler, J. M., Autenrieth, M., & Beluch, W. H. (2019). *Uncertainty Based Detection and Relabeling of Noisy Image Labels*. http://arxiv.org/abs/1906.11876

Kuenzer, C., Guo, H., Schlegel, I., Tuan, V., Li, X., & Dech, S. (2013). Varying Scale and Capability of Envisat ASAR-WSM, TerraSAR-X Scansar and TerraSAR-X Stripmap Data to Assess Urban Flood Situations: A Case Study of the Mekong Delta in Can Tho Province. *Remote Sensing*, *5*(10), 5122–5142. https://doi.org/10.3390/rs5105122

Lavalle, M., Simard, M., & Hensley, S. (2011). A temporal decorrelation model for polarimetric radar interferometers. *IEEE Transactions on Geoscience and Remote Sensing*, *50*(7), 2880-2888.

LaBonte, T., Martinez, C., & Roberts, S. A. (2019). *We Know Where We Don't Know: 3D Bayesian CNNs for Credible Geometric Uncertainty*. http://arxiv.org/abs/1910.10793

Lakshmi, V. (Ed.). (2017). *Remote Sensing of Hydrological Extremes*. Springer International Publishing. https://doi.org/10.1007/978-3-319-43744-6

Li, K., Wan, G., Cheng, G., Meng, L., & Han, J. (2020). 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/10.1016/j.isprsjprs.2019.11.023

Li, Z., Wang, C., Emrich, C. T., & Guo, D. (2017). A novel approach to leveraging social media for rapid flood mapping: a case study of the 2015 South Carolina floods. *Https://Doi.Org/10.1080/15230406.2016.1271356*, *45*(2), 97–110. https://doi.org/10.1080/15230406.2016.1271356

Liu, B., Li, X., & Zheng, G. (2019). Coastal Inundation Mapping From Bitemporal and Dual-Polarization SAR Imagery Based on Deep Convolutional Neural Networks. *Journal of Geophysical Research: Oceans*, *124*(12), 9101–9113. https://doi.org/10.1029/2019JC015577

Liu, T., Li, Y., Cao, Y., & Shen, Q. (2017). Change detection in multitemporal synthetic aperture radar images using dual-channel convolutional neural network. *Journal of Applied Remote Sensing*, *11*(04), 1. https://doi.org/10.1117/1.jrs.11.042615

Manavalan, R. (2017). SAR image analysis techniques for flood area mapping - literature survey. *Earth Science Informatics*, *10*(1), 1–14. https://doi.org/10.1007/s12145-016-0274-2

Manavalan, R., & Ramanuja. (2018). Review of synthetic aperture radar frequency, polarization, and incidence angle data for mapping the inundated regions. *Journal of Applied Remote Sensing*, *12*(02), 1. https://doi.org/10.1117/1.jrs.12.021501

Marconcini, M., Metz-Marconcini, A., Üreyen, S., Palacios-Lopez, D., Hanke, W., Bachofer, F., Zeidler, J., Esch, T., Gorelick, N., Kakarla, A., & Strano, E. (2019). *Outlining where humans live -- The World Settlement Footprint 2015*.

Markert, K. N., Chishtie, F., Anderson, E. R., Saah, D., & Griffin, R. E. (2018). On the merging of optical and SAR satellite imagery for surface water mapping applications. *Results in Physics*, *9*, 275–277. https://doi.org/10.1016/j.rinp.2018.02.054

Martin, T., Avleen, B., Peter, R., & Nathan, S. (2013). Mini-batch primal and dual methods for SVMs. *International Conference on Machine Learning*.

Martinis, S., Kuenzer, C., & Twele, A. (2015). Flood studies using Synthetic Aperture Radar data. In P. Thenkabail (Ed.), *Remote Sensing Handbook Volume III - Remote Sensing of Water Resources, Disasters, and Urban Studies* (Vol. 3, pp. 145–173). Taylor and Francis.

Martinis, S., Kuenzer, C., Wendleder, A., Huth, J., Twele, A., Roth, A., & Dech, S. (2015). Comparing four operational SAR-based water and flood detection approaches. *International Journal of Remote Sensing*, *36*(13), 3519–3543. https://doi.org/10.1080/01431161.2015.1060647

Martinis, S., Plank, S., & Ćwik, K. (2018). The Use of Sentinel-1 Time-Series Data to Improve Flood Monitoring in Arid Areas. *Remote Sensing*, *10*(4), 583. https://doi.org/10.3390/rs10040583

Martinis, S., Twele, A., & Voigt, S. (2009). Towards operational near real-time flood detection using a split-based automatic thresholding procedure on high resolution TerraSAR-X data. *Natural Hazards and Earth System Sciences*, *9*(2), 303–314. https://doi.org/10.5194/nhess-9-303-2009

Matgen, P., Hostache, R., Schumann, G., Pfister, L., Hoffmann, L., & Savenije, H. H. G. (2011). Towards an automated SAR-based flood monitoring system: Lessons learned from two case studies. *Physics and Chemistry of the Earth, Parts A/B/C*, *36*(7–8), 241–252. https://doi.org/10.1016/j.pce.2010.12.009

McGill, R., Tukey, J. W., & Larsen, W. A. (1978). Variations of box plots. *American Statistician*, *32*(1), 12–16. https://doi.org/10.1080/00031305.1978.10479236

Mederer, P. (2020). *Die Kombination von Fernerkundung- und In-Situ-Daten zur Optimierung des Lagebildes bei Katastrophen*. KU Eichstätt-Ingolstadt.

Melnyk, R., & Shokur, Y. L. (2016). Image compression based on the Visvalingam-Whyatt algorithm. *Undefined*.

Merz, B., Kreibich, H., Schwarze, R., & Thieken, A. (2010). Review article "Assessment of economic flood damage." *Natural Hazards and Earth System Sciences*, *10*(8), 1697–1724. https://doi.org/10.5194/NHESS-10-1697-2010

Moore, G. K., & North, G. W. (1974). Flood Inundation in the Southeastern United States from Aircraft and Satellite Imagery. *JAWRA Journal of the American Water Resources Association*, *10*(5), 1082–1096. https://doi.org/10.1111/j.1752-1688.1974.tb00626.x

Mudashiru, R. B., Sabtu, N., Abustan, I., & Balogun, W. (2021). Flood hazard mapping methods: A review. *Journal of Hydrology*, *603*, 126846. https://doi.org/10.1016/J.JHYDROL.2021.126846

Munasinghe, D., Cohen, S., Huang, Y.-F., Tsang, Y.-P., Zhang, J., & Fang, Z. (2018). Intercomparison of Satellite Remote Sensing-Based Flood Inundation Mapping Techniques. *JAWRA Journal of the American Water Resources Association*, *54*(4), 834–846. https://doi.org/10.1111/1752-1688.12626

Muñoz, D. F., Muñoz, P., Moftakhari, H., & Moradkhani, H. (2021). From local to regional compound flood mapping with deep learning and data fusion techniques. *Science of The Total Environment*, *782*, 146927. https://doi.org/10.1016/j.scitotenv.2021.146927

Mustafa, E. M., Fouad, M. M., & Elshafey, M. A. (2019). Enhancing the Performance of an Image Steganalysis Approach Using Variable Batch Size-Based CNN on GPUs. *2019 10th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, 40–44. https://doi.org/10.1109/IDAACS.2019.8924348

Nemni, E., Bullock, J., Belabbes, S., & Bromley, L. (2020). Fully Convolutional Neural Network for Rapid Flood Segmentation in Synthetic Aperture Radar Imagery. *Remote Sensing*, *12*(16), 2532. https://doi.org/10.3390/rs12162532

O'Grady, D., Leblanc, M., & Gillieson, D. (2011). Use of ENVISAT ASAR Global Monitoring Mode to complement optical data in the mapping of rapid broad-scale flooding in Pakistan. *Hydrology and Earth System Sciences*, *15*(11), 3475–3494. https://doi.org/10.5194/hess-15-3475-2011

Olson, D. M., Dinerstein, E., Wikramanayake, E. D., Burgess, N. D., … (2001). Terrestrial ecoregions of the world: A new map of life on Earth. In *BioScience* (Vol. 51, Issue 11, pp. 933–938). Oxford Academic. https://doi.org/10.1641/0006-3568(2001)051[0933:TEOTWA]2.0.CO;2

Otsu, N. (1979). A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, *9*(1), 62–66. https://doi.org/10.1109/TSMC.1979.4310076

Pai, M. M. M., Mehrotra, V., Verma, U., & Pai, R. M. (2020). Improved Semantic Segmentation of Water Bodies and Land in SAR Images Using Generative Adversarial Networks. *International Journal of Semantic Computing*, *14*(01), 55–69. https://doi.org/10.1142/S1793351X20400036

Pekel, J.-F., Cottam, A., Gorelick, N., & Belward, A. S. (2016). High-resolution mapping of global surface water and its long-term changes. *Nature*, *540*(7633), 418–422. https://doi.org/10.1038/nature20584

QGIS Development Team. (2022). *QGIS Geographic Information System.* Open Source Geospatial Foundation Project. . http://qgis.osgeo.org/ (*accessed 08-14-2022*)

Redekop, E., & Chernyavskiy, A. (2021). Uncertainty-based method for improving poorly labeled segmentation datasets. *18th International Symposium on Biomedical Imaging (ISBI)*, 1831–1835.

Ren, M., Zeng, W., Yang, B., & Urtasun, R. (2018). Learning to reweight examples for robust deep learning. . *International Conference on Machine Learning*, 4334–4343.

Rigge, M., Homer, C., Shi, H., Meyer, D. K., Bunde, B., Granneman, B., Postma, K., Danielson, P., Case, A., & Xian, G. (2021). Rangeland fractional components across the western United States from 1985 to 2018. *Remote Sensing*, *13*(4), 1–26. https://doi.org/10.3390/rs13040813

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, *323*(6088), 533–536. https://doi.org/10.1038/323533a0

Sanyal, J., & Lu, X. X. (2004a). Application of remote sensing in flood management with special reference to monsoon Asia: A review. In Natural Hazards (Vol. 33, Issue 2, pp. 283–301). Springer. https://doi.org/10.1023/B:NHAZ.0000037035.65105.95

Schavemaker, J. G. M., Reinders, M. J. T., Gerbrands, J. J., & Backer, E. (2000). Image sharpening by morphological filtering. *Pattern Recognition*, *33*(6), 997–1012. https://doi.org/10.1016/S0031-3203(99)00160-0

Shen, X., Wang, D., Mao, K., Anagnostou, E., & Hong, Y. (2019). Inundation Extent Mapping by Synthetic Aperture Radar: A Review. *Remote Sensing*, *11*(7), 879. https://doi.org/10.3390/rs11070879

Shridhar, K., Laumann, F., & Liwicki, M. (2019). *A Comprehensive guide to Bayesian Convolutional Neural Network with Variational Inference*. http://arxiv.org/abs/1901.02731

Simonyan, K., & Zisserman, A. (2014). *Very Deep Convolutional Networks for Large-Scale Image Recognition*. https://doi.org/10.48550/arxiv.1409.1556

Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, *45*(4), 427–437. https://doi.org/10.1016/j.ipm.2009.03.002

Sountov, P., Suter, C., Burnim, J., Dillon, J., & Tensorflow Probability team. (2019). *Regression with probabilistic layers in tensorflow probability. .* https://blog.tensorflow.org/2019/03/regression-with-probabilistic-layers-in.html

Story, M., & Congalton, R. (1986). Accuracy assessment: a user's perspective. *Photogrammetric Engineering and Remote Sensing*, *52*(3), 397–399.

Sumaiya, M. N., & Shantha Selva Kumari, R. (2018). Unsupervised change detection of flood affected areas in SAR images using Rayleigh-based Bayesian thresholding. *IET Radar, Sonar & Navigation*, *12*(5), 515–522. https://doi.org/10.1049/iet-rsn.2017.0393

Suwarsono, Nugroho, J. T., & Wiweka. (2013). Identification of inundated area using normalized difference water index (NDWI) on lowland region of java island. *34th Asian Conference on Remote Sensing 2013, ACRS 2013*, *4*(2), 3783–3789. https://doi.org/10.30536/j.ijreses.2013.v10.a1850

Tabari, H. (2020). Climate change impact on flood and extreme precipitation increases with water availability. *Scientific Reports*, *10*(1), 13768. https://doi.org/10.1038/s41598-020-70816-2

Tsakiris, G. (2014). Flood risk assessment: Concepts, modelling, applications. *Natural Hazards and Earth System Sciences*, *14*(5), 1361–1369. https://doi.org/10.5194/NHESS-14-1361-2014

Twele, A., Cao, W., Plank, S., & Martinis, S. (2016). Sentinel-1-based flood mapping: a fully automated processing chain. *International Journal of Remote Sensing*, *37*(13), 2990–3004. https://doi.org/10.1080/01431161.2016.1192304

van de Schoot, R., Depaoli, S., King, R., Kramer, B., Märtens, K., Tadesse, M. G., Vannucci, M., Gelman, A., Veen, D., Willemsen, J., & Yau, C. (2021). Bayesian statistics and modelling. *Nature Reviews Methods Primers*, *1*(1), 1. https://doi.org/10.1038/s43586-020-00001-2

van der Aalst, W. M. P., Rubin, V., Verbeek, H. M. W., van Dongen, B. F., Kindler, E., & Günther, C. W. (2010). Process mining: a two-step approach to balance between underfitting and overfitting. *Software & Systems Modeling*, *9*(1), 87–111. https://doi.org/10.1007/s10270-008-0106-z

van der Sande, C. J., de Jong, S. M., & de Roo, A. P. J. (2003). A segmentation and classification approach of IKONOS-2 imagery for land cover mapping to assist flood risk and flood damage assessment. *International Journal of Applied Earth Observation and Geoinformation*, *4*(3), 217–229. https://doi.org/10.1016/S0303-2434(03)00003-5

van Rossum, G. (1995). Python reference manual. *Department of Computer Science [CS]*, *R 9525*.

Visvalingam, M., & Whyatt, J. D. (1993). Line generalisation by repeated elimination of points. *The Cartographic Journal*, *30*(1), 46–51. https://doi.org/10.1179/000870493786962263

Wang, S.-C. (2003). Artificial Neural Network. In *Interdisciplinary Computing in Java Programming* (pp. 81–100). Springer US. https://doi.org/10.1007/978-1-4615-0377-4_5

Wei, Z., & Chen, X. (2021). Uncertainty Quantification in Inverse Scattering Problems With Bayesian Convolutional Neural Networks. *IEEE Transactions on Antennas and Propagation*, *69*(6), 3409–3418. https://doi.org/10.1109/TAP.2020.3030974

Wen, Y., Vicol, P., Ba, J., Tran, D., & Grosse, R. (2018). *Flipout: Efficient Pseudo-Independent Weight Perturbations on Mini-Batches*. https://doi.org/10.48550/arxiv.1803.04386

Westerhoff, R. S., Kleuskens, M. P. H., Winsemius, H. C., Huizinga, H. J., Brakenridge, G. R., & Bishop, C. (2013). Automated global water mapping based on wide-swath orbital synthetic-aperture radar. *Hydrology and Earth System Sciences*, *17*(2), 651–663. https://doi.org/10.5194/hess-17-651-2013

Wieland, M., Helleis, M., Krullikowski, C., Martinis, S., & Plank, S. (2022). *Data_s1s2_water: A global dataset for semantic segmentation of water bodies from Sentinel-1 and Sentinel-2 data.*

Wieland, M., & Martinis, S. (2019). A Modular Processing Chain for Automated Flood Monitoring from Multi-Spectral Satellite Data. *Remote Sensing*, *11*(19), 2330. https://doi.org/10.3390/rs11192330

Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional neural networks: an overview and application in radiology. *Insights into Imaging*, *9*(4), 611–629. https://doi.org/10.1007/s13244-018-0639-9

You, K., Long, M., Wang, J., & Jordan, M. I. (2019). How Does Learning Rate Decay Help Modern Neural Networks? *ArXiv Preprint*.

Zhai, Y., Qu, Z., & Hao, L. (2018). Land Cover Classification Using Integrated Spectral, Temporal, and Spatial Features Derived from Remotely Sensed Images. *Remote Sensing*, *10*(3), 383. https://doi.org/10.3390/rs10030383

Appendix

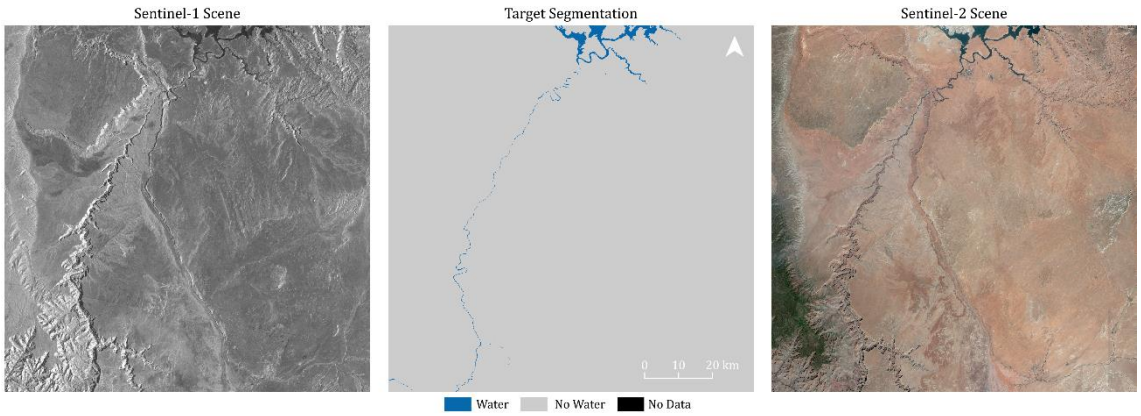
A Test Scene Overview

A.1 Table containing basic information about all 18 test scenes

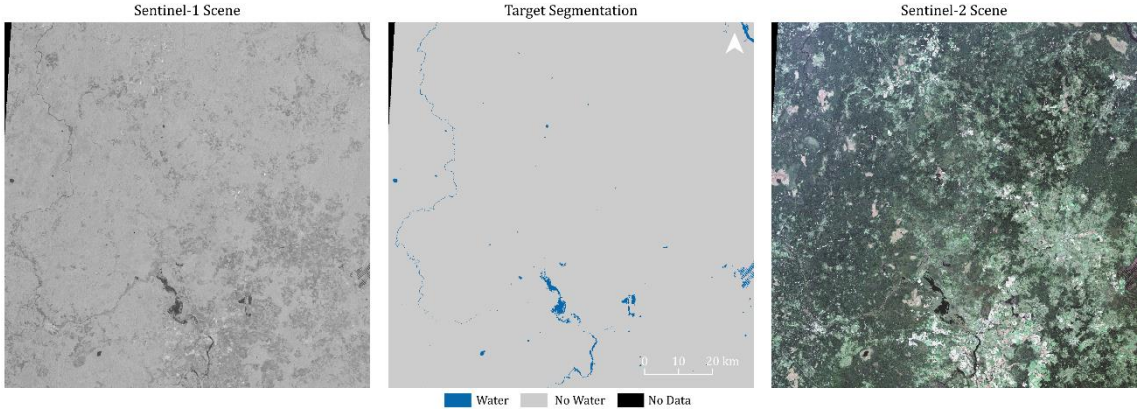
Scene ID	Valid Pixels	Main Landcover	Acquisition Date
23	148,840,000	Bare	2019-08-18
25	331,838,173	Closed Forest	2019-08-27
29	148,789,852	Closed Forest	2019-09-02
31	14,263,341	Herb, Vegetation	2019-09-10
33	188,234,319	Closed Forest	2019-09-18
35	176,448,793	Closed Forest	2019-09-16
47	82,043,578	Herb, Vegetation	2019-01-04
53	188,190,103	Cultivated Vegetation	2019-04-16
57	181,540,054	Permanent Water	2019-06-17
66	135,182,112	Shrubs	2019-08-29
75	188,259,427	Cultivated Vegetation	2019-09-11
77	61,804,734	Closed Forest	2019-09-18
78	172,199,461	Herb, Vegetation	2019-09-16
80	138,390,797	Herb, Vegetation	2019-10-14
82	148,834,722	Closed Forest	2019-09-26
88	129,307,897	Bare	2020-11-24
89	137,534,256	Sea	2020-04-10
90	133,872,887	Closed Forest	2020-10-13

A.2 Maps of all 18 test scenes with Sentinel-1 and Sentinel-2 and the reference data

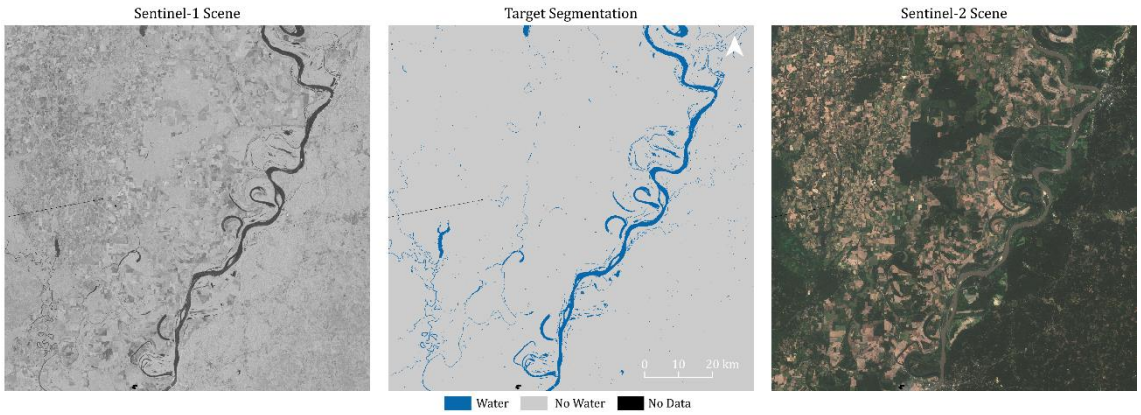
Scene 23



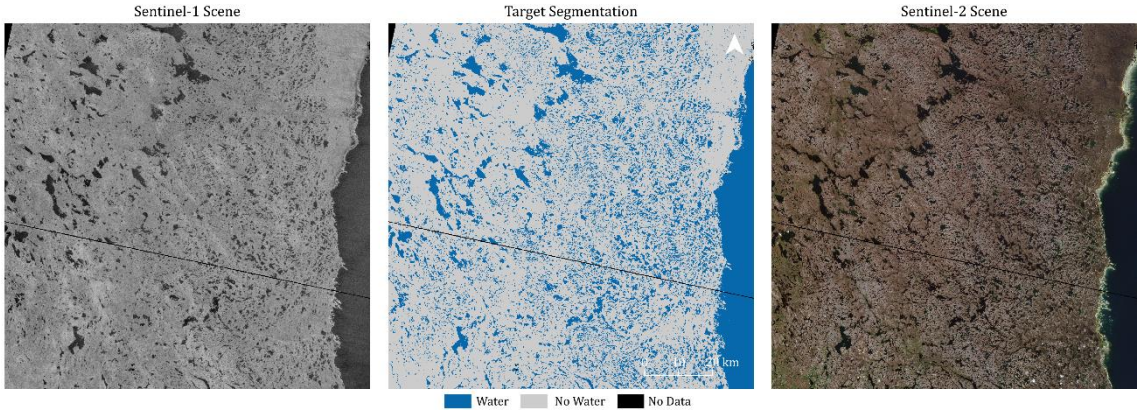
Scene 25



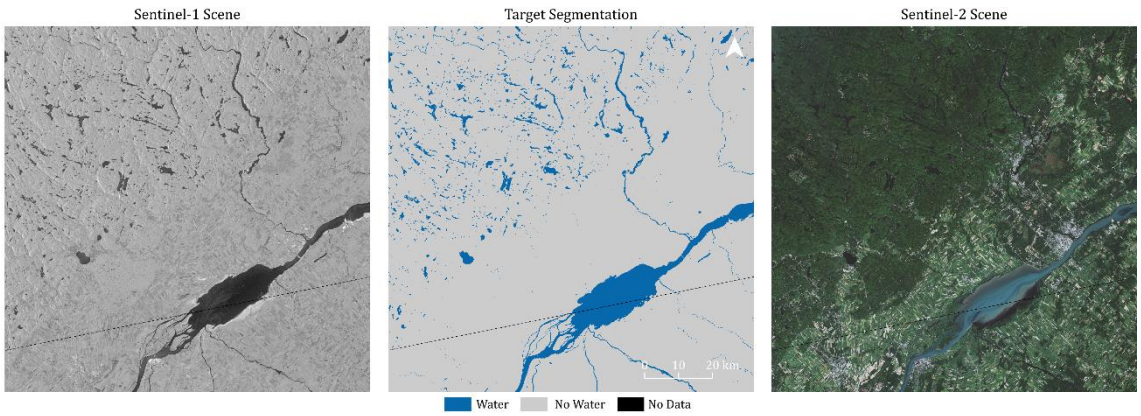
Scene 29



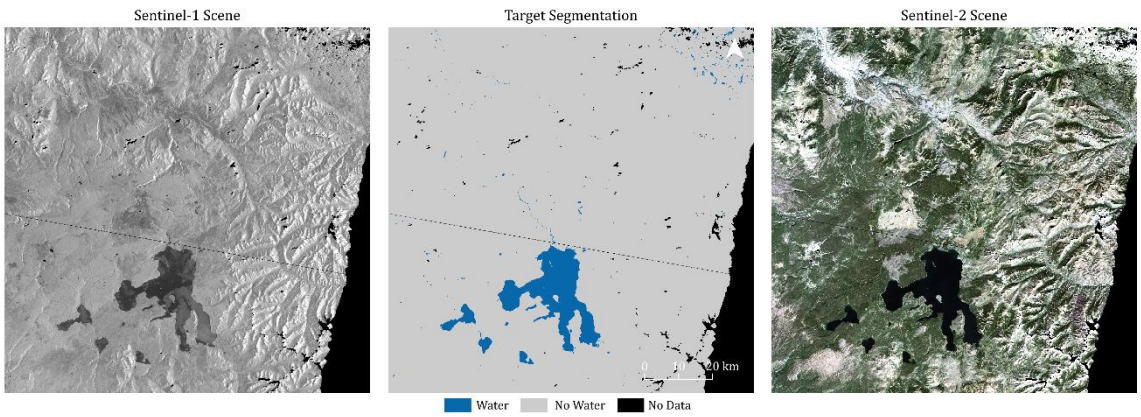
Scene 31



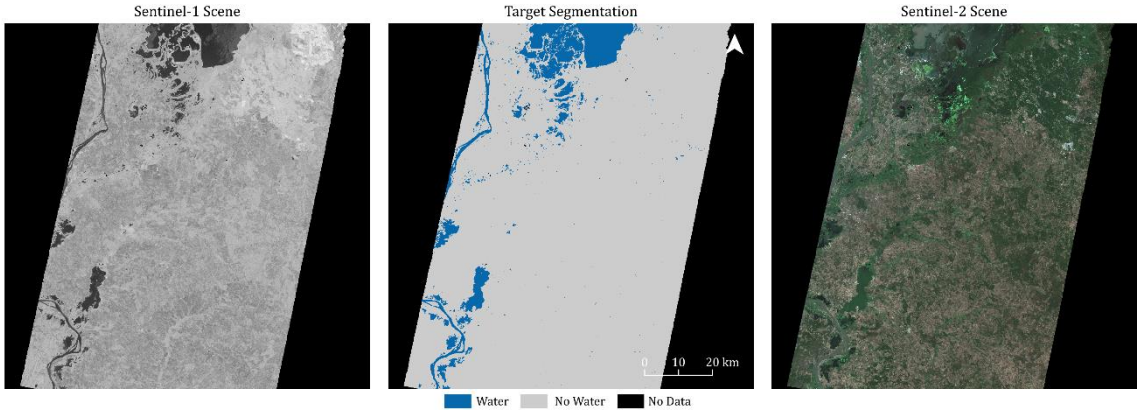
Scene 33



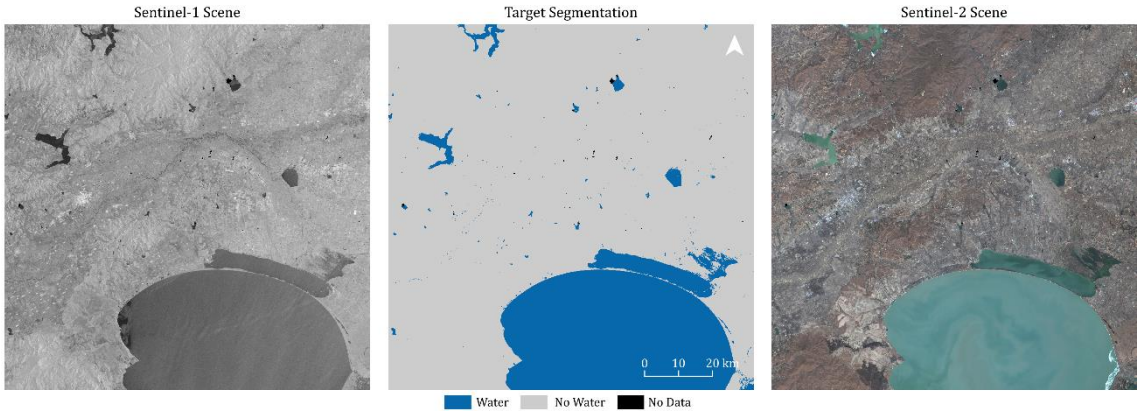
Scene 35



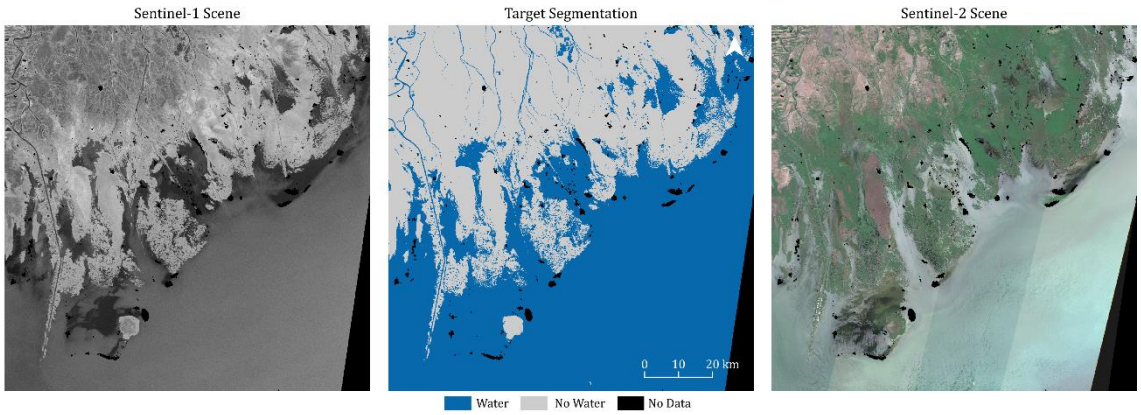
Scene 47



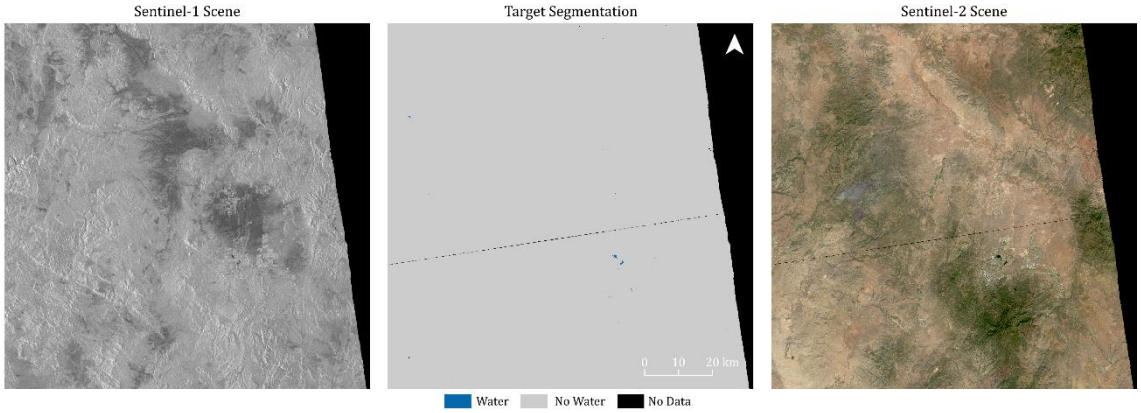
Scene 53



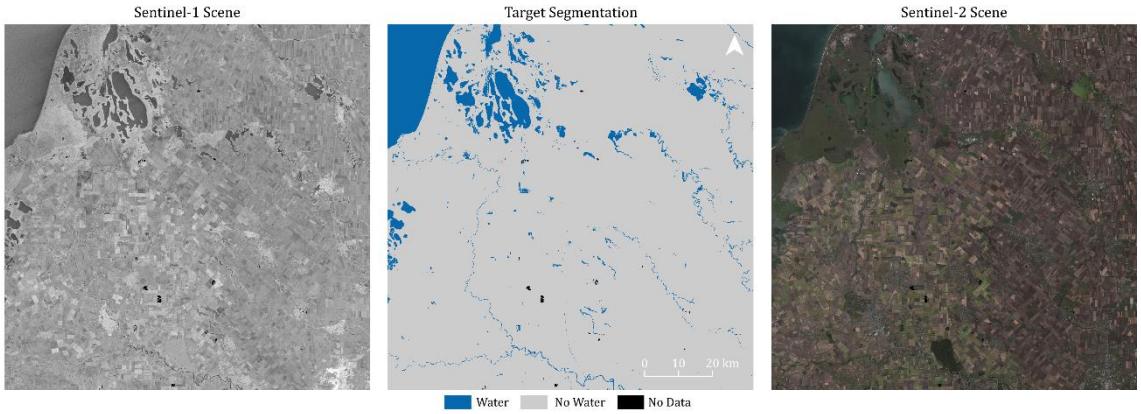
Scene 57



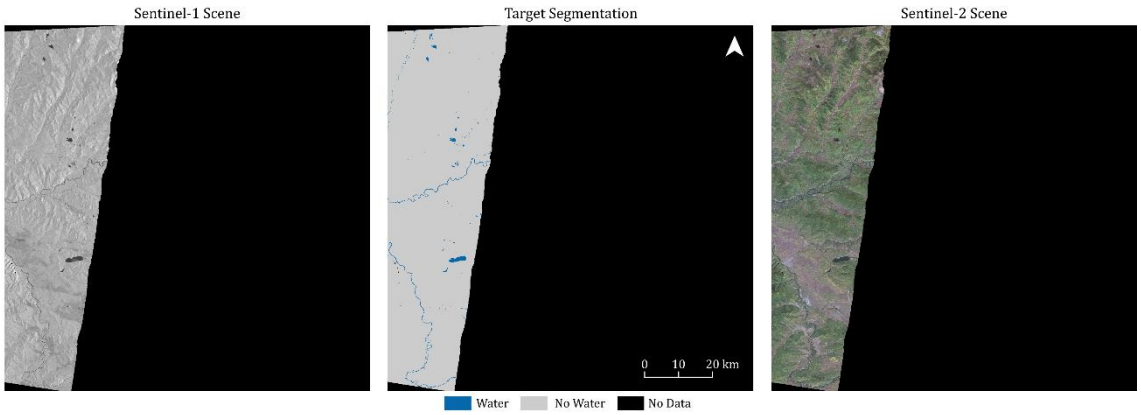
Scene 66



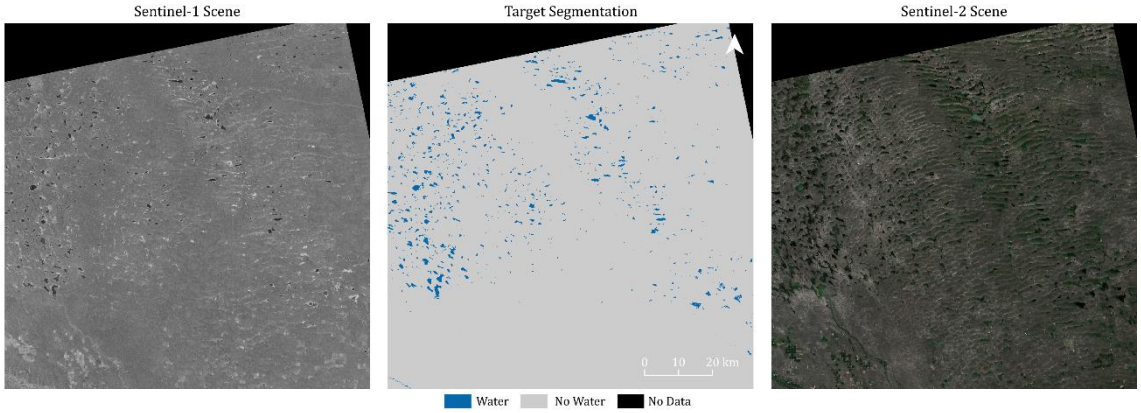
Scene 75



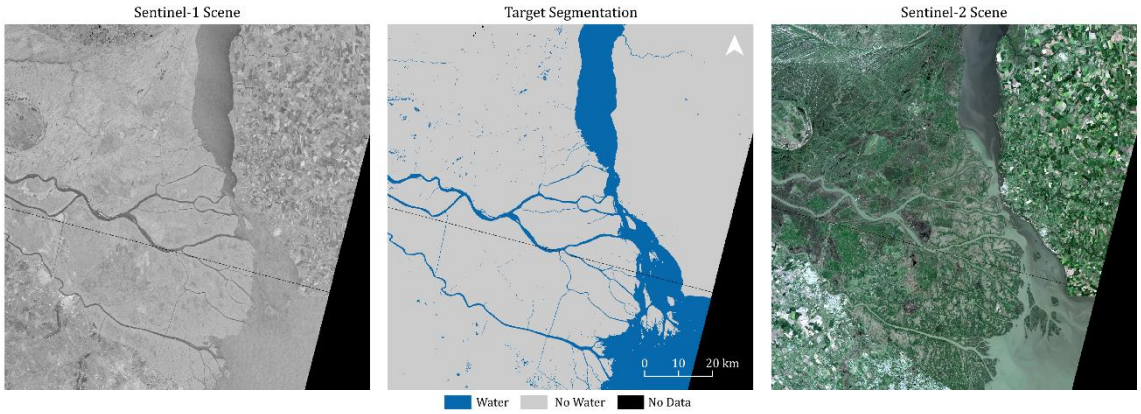
Scene 77



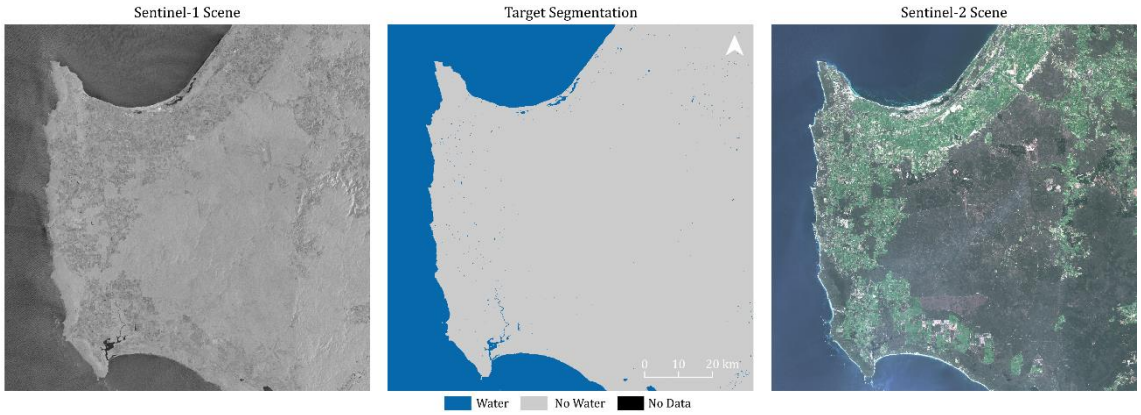
Scene 78



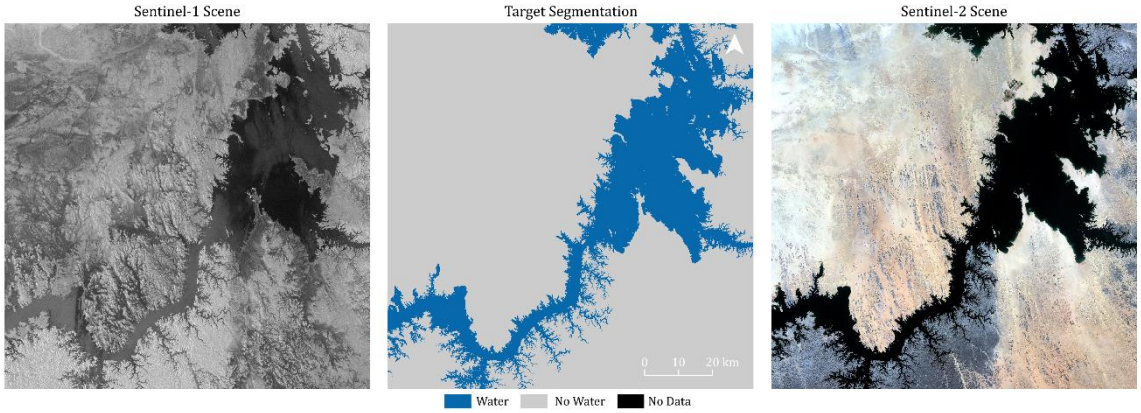
Scene 80



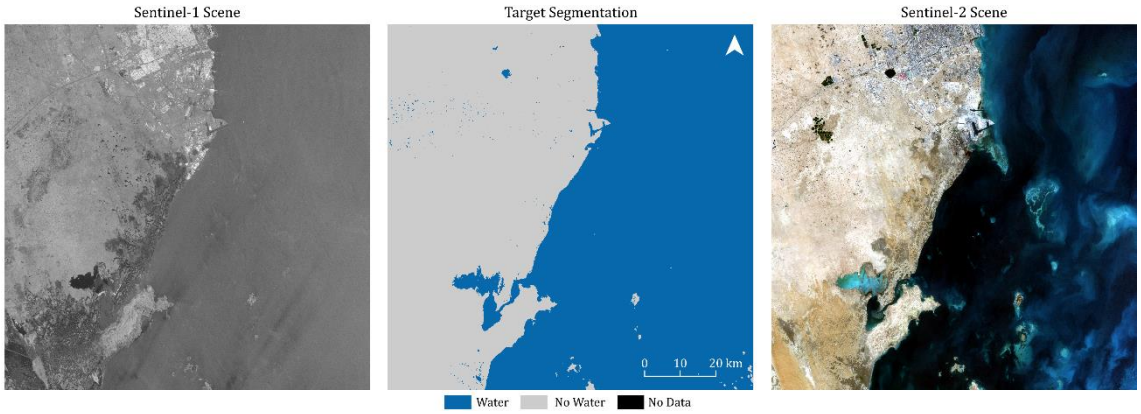
Scene 82



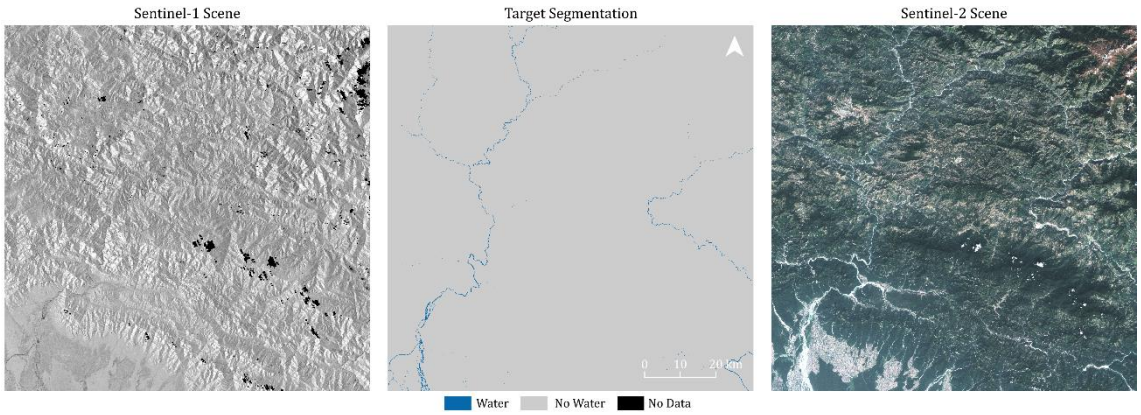
Scene 88



Scene 89

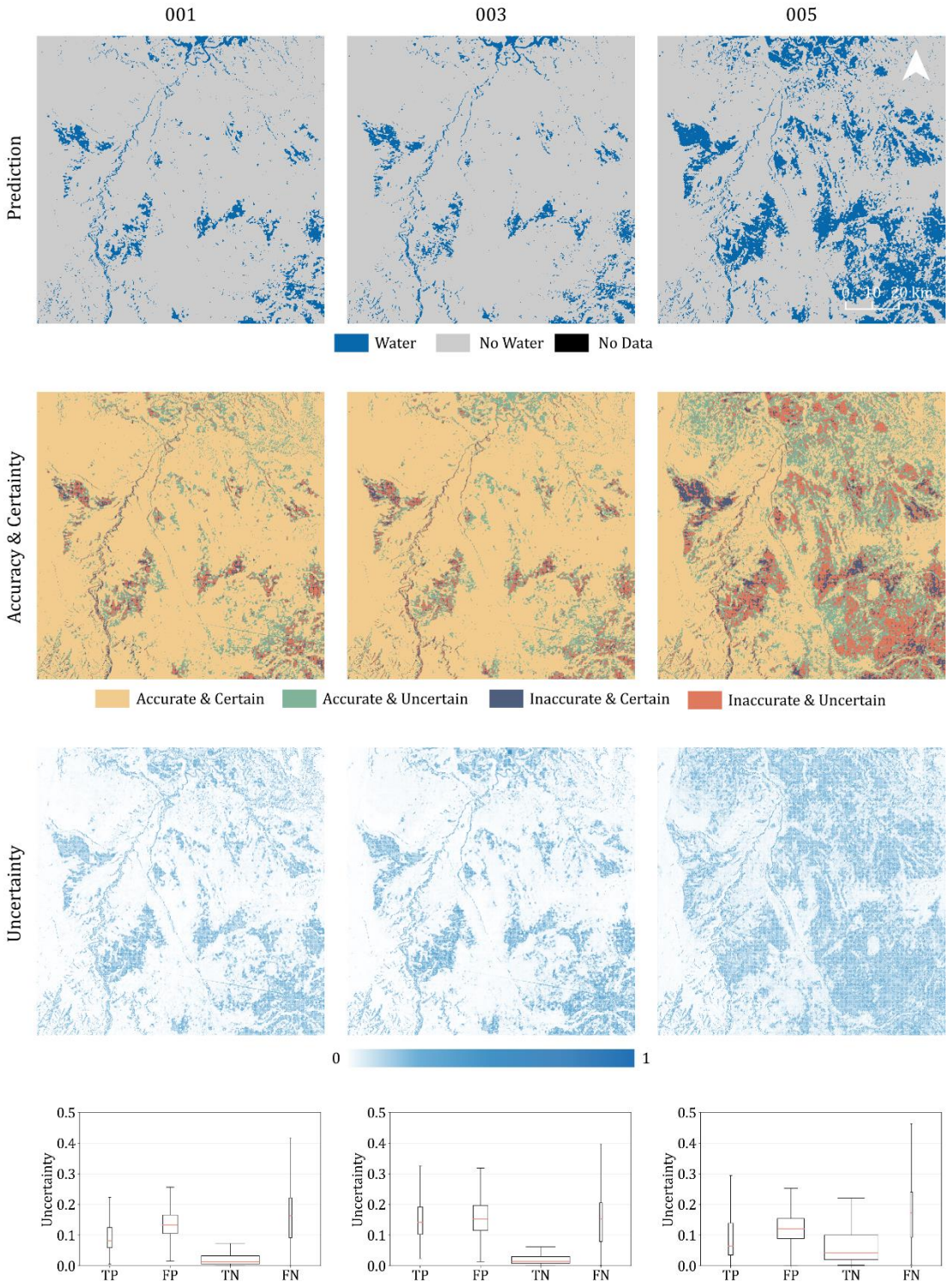


Scene 90

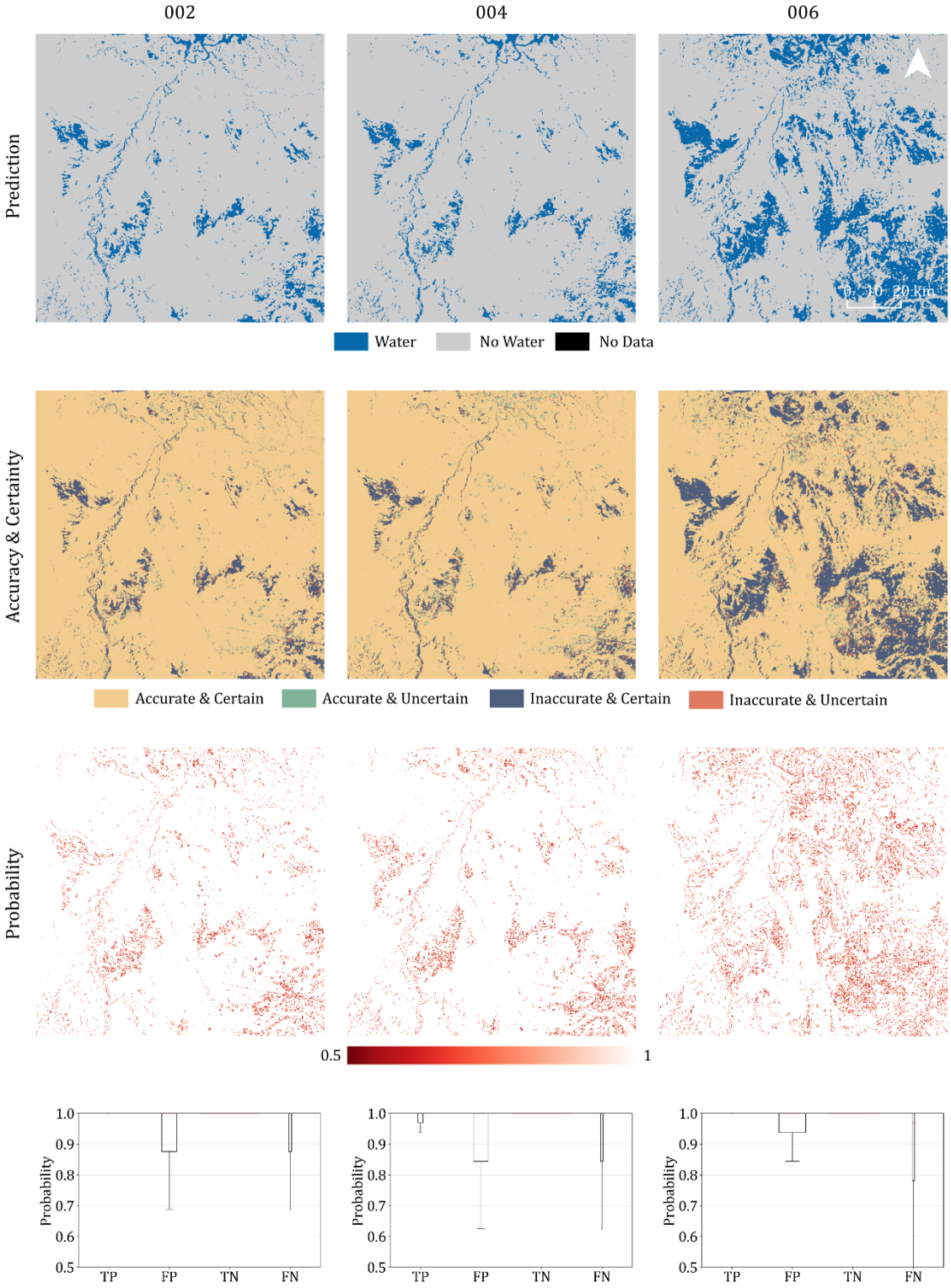


B Accuracy and Certainty Maps

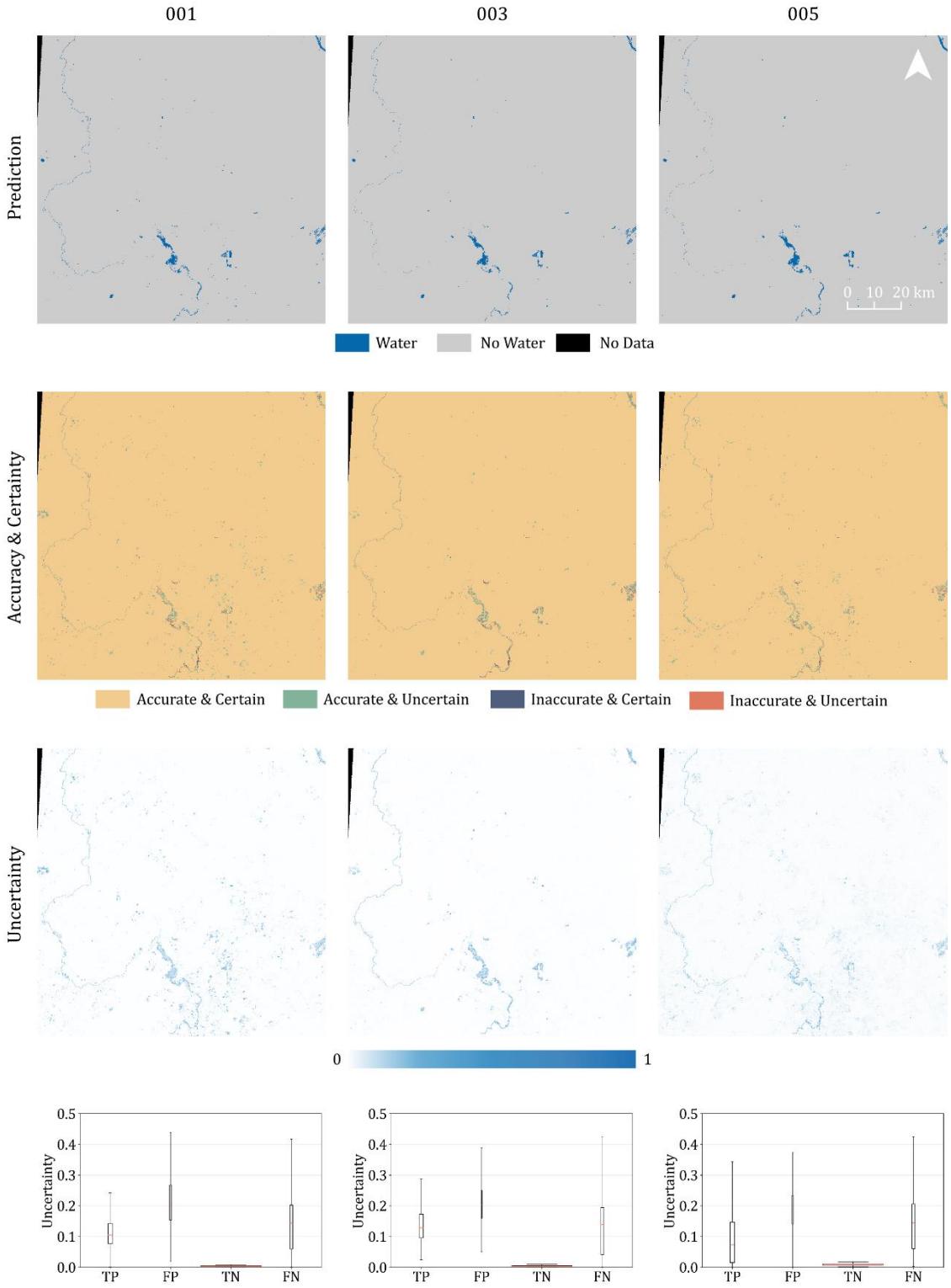
Scene 23, UD1:



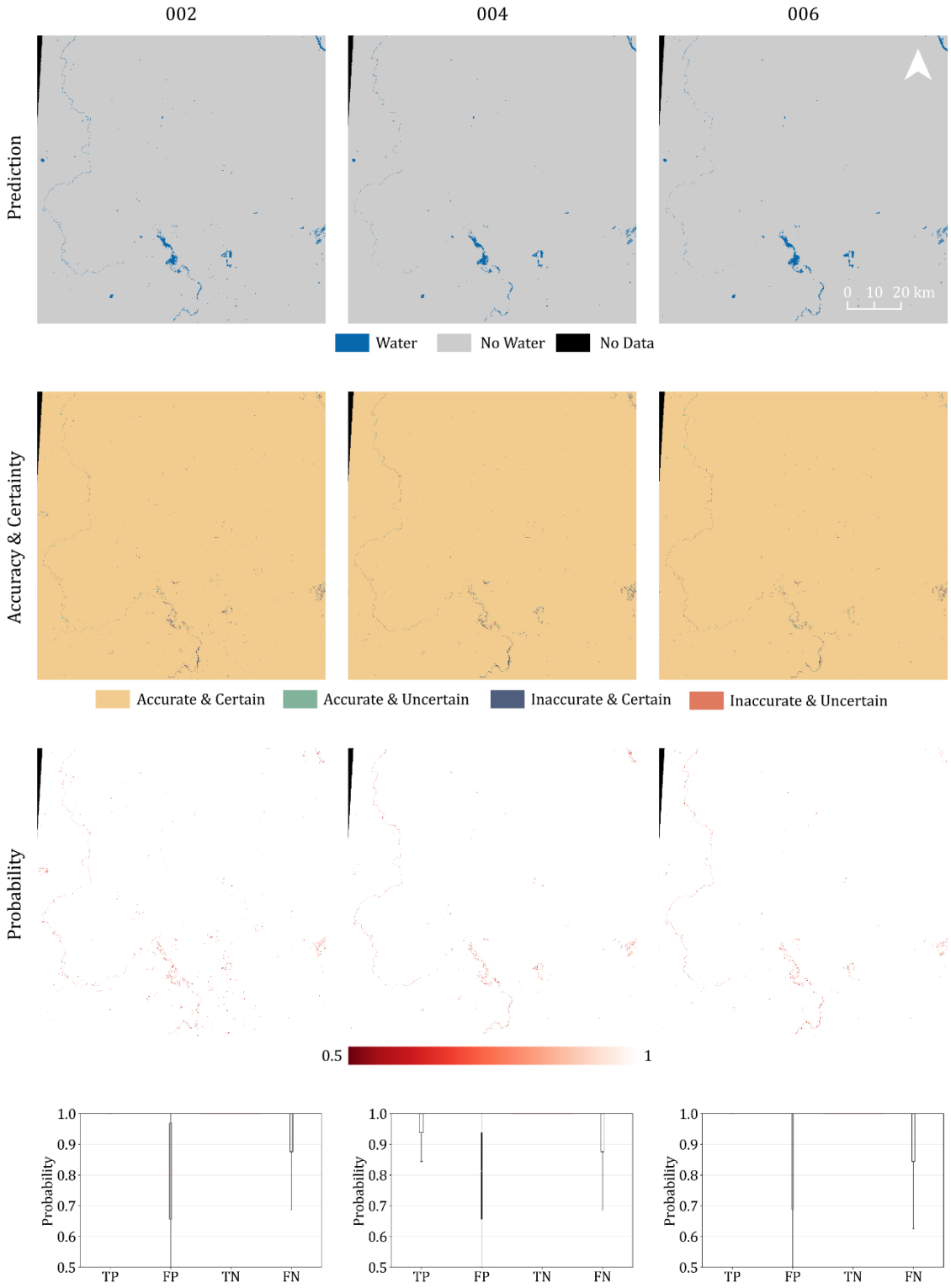
Scene 23, UD2:


002
004
006
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Probability
0.5
1
Probability
1.0
0.9
0.8
0.7
0.6
0.5
TP
FP
TN
FN

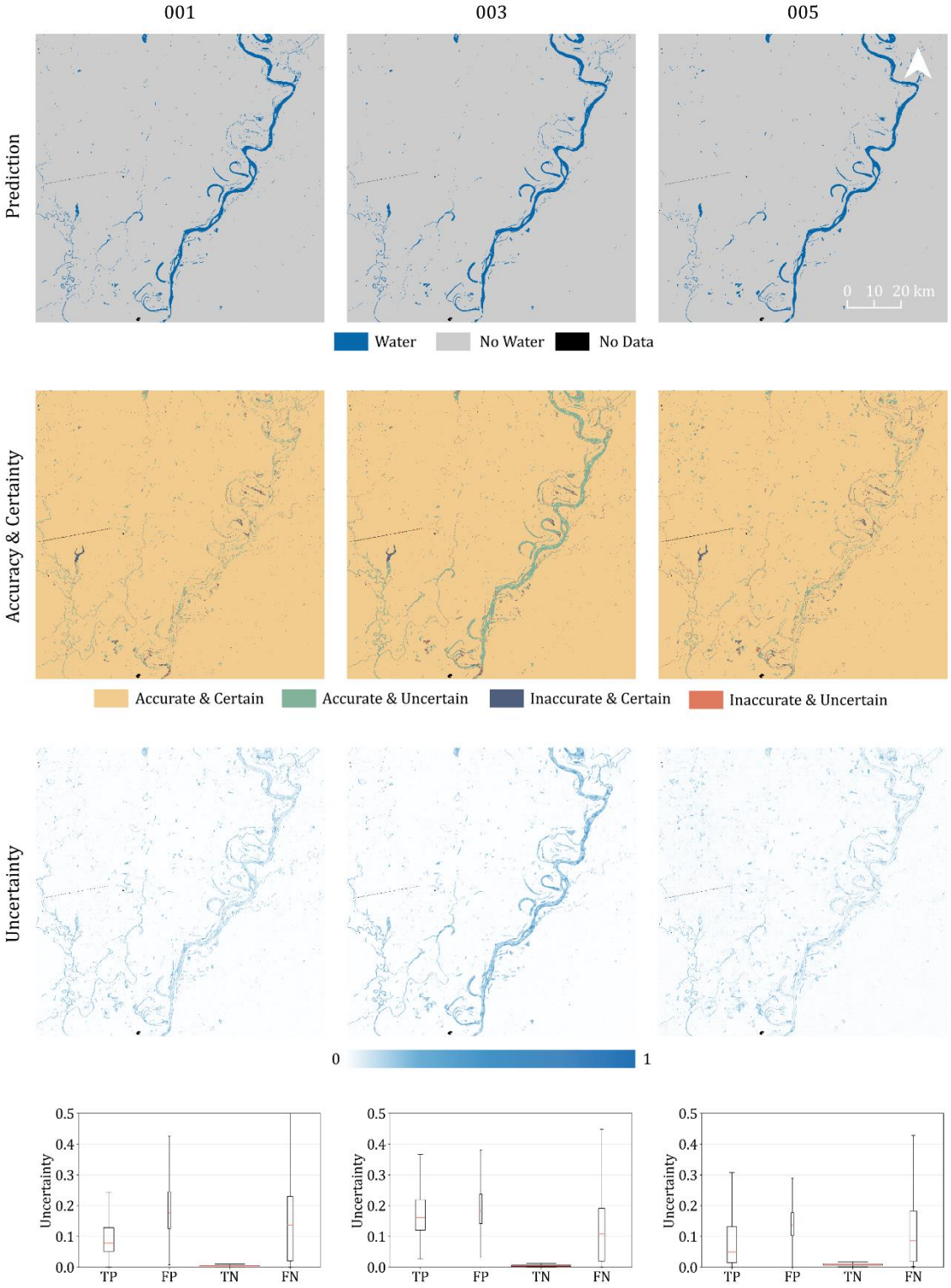
Scene 25, UD1:


001
003
005
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Uncertainty
0
1
Uncertainty
0.0
0.1
0.2
0.3
0.4
0.5
TP
FP
TN
FN

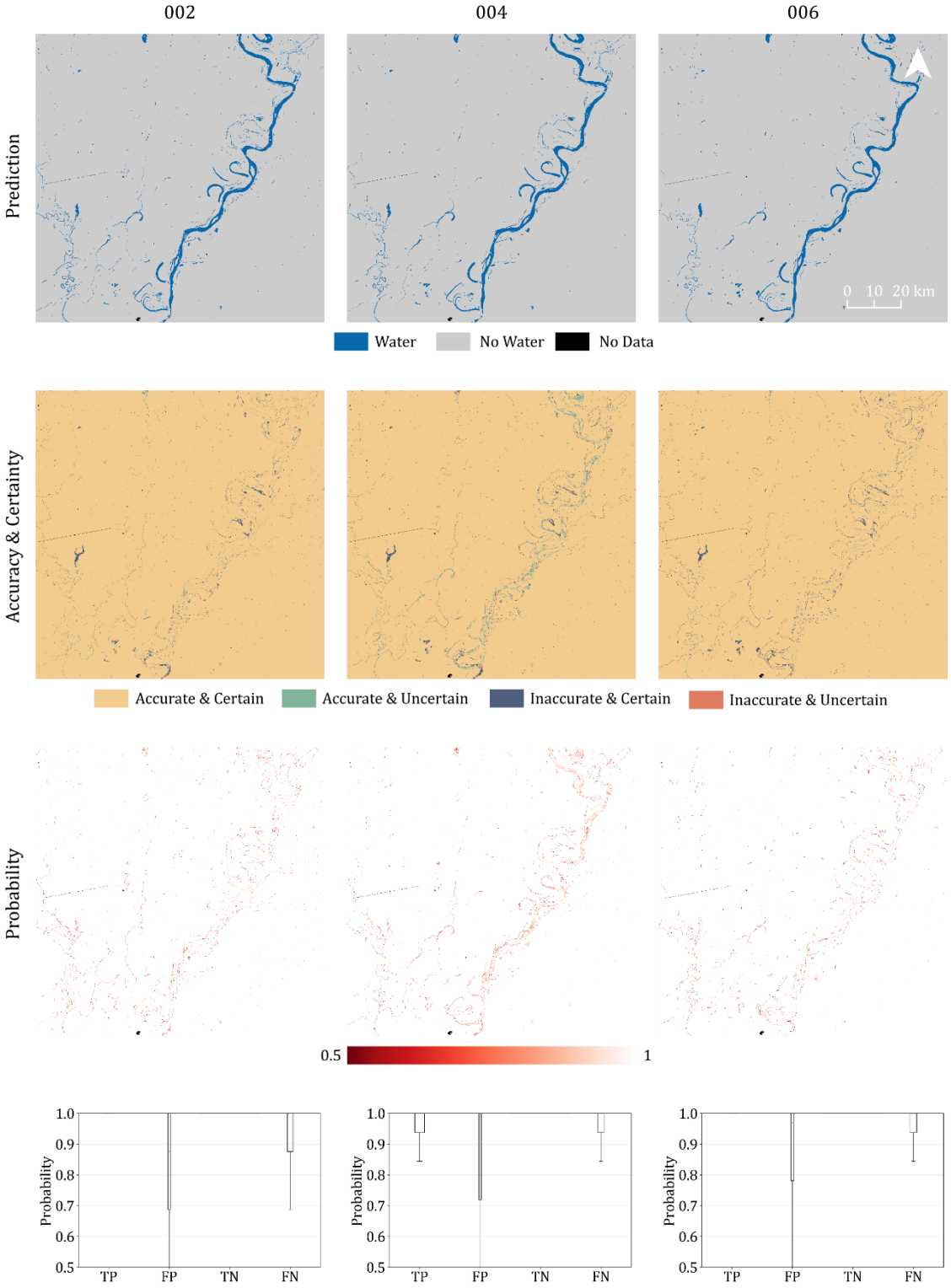
Scene 25, UD2:


002
004
006
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Probability
0.5
1
Probability
1.0
0.9
0.8
0.7
0.6
0.5
TP
FP
TN
FN

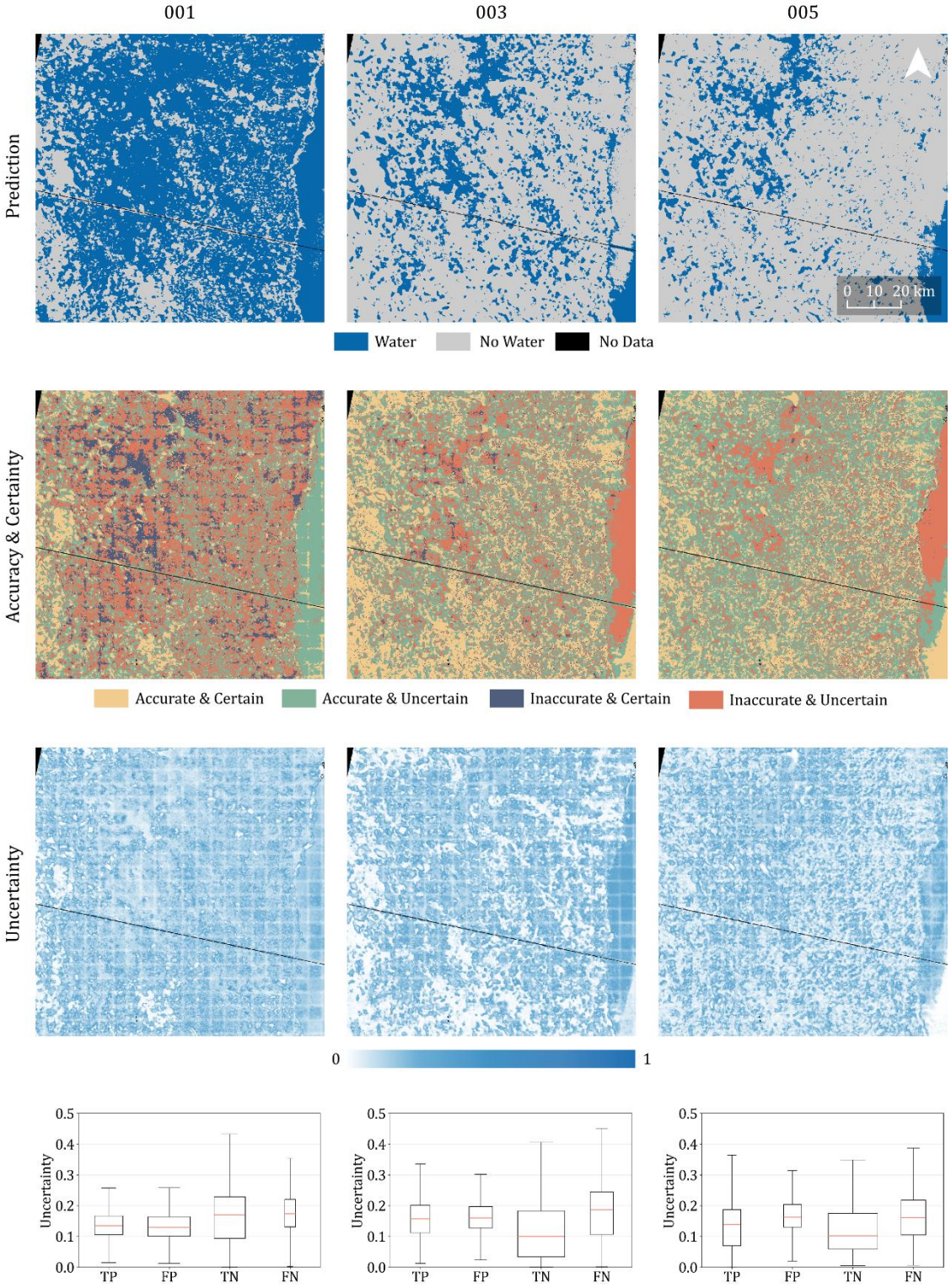
Scene 29, UD1:


001
003
005
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Uncertainty
0
1
Uncertainty
0.0
0.1
0.2
0.3
0.4
0.5
TP
FP
TN
FN

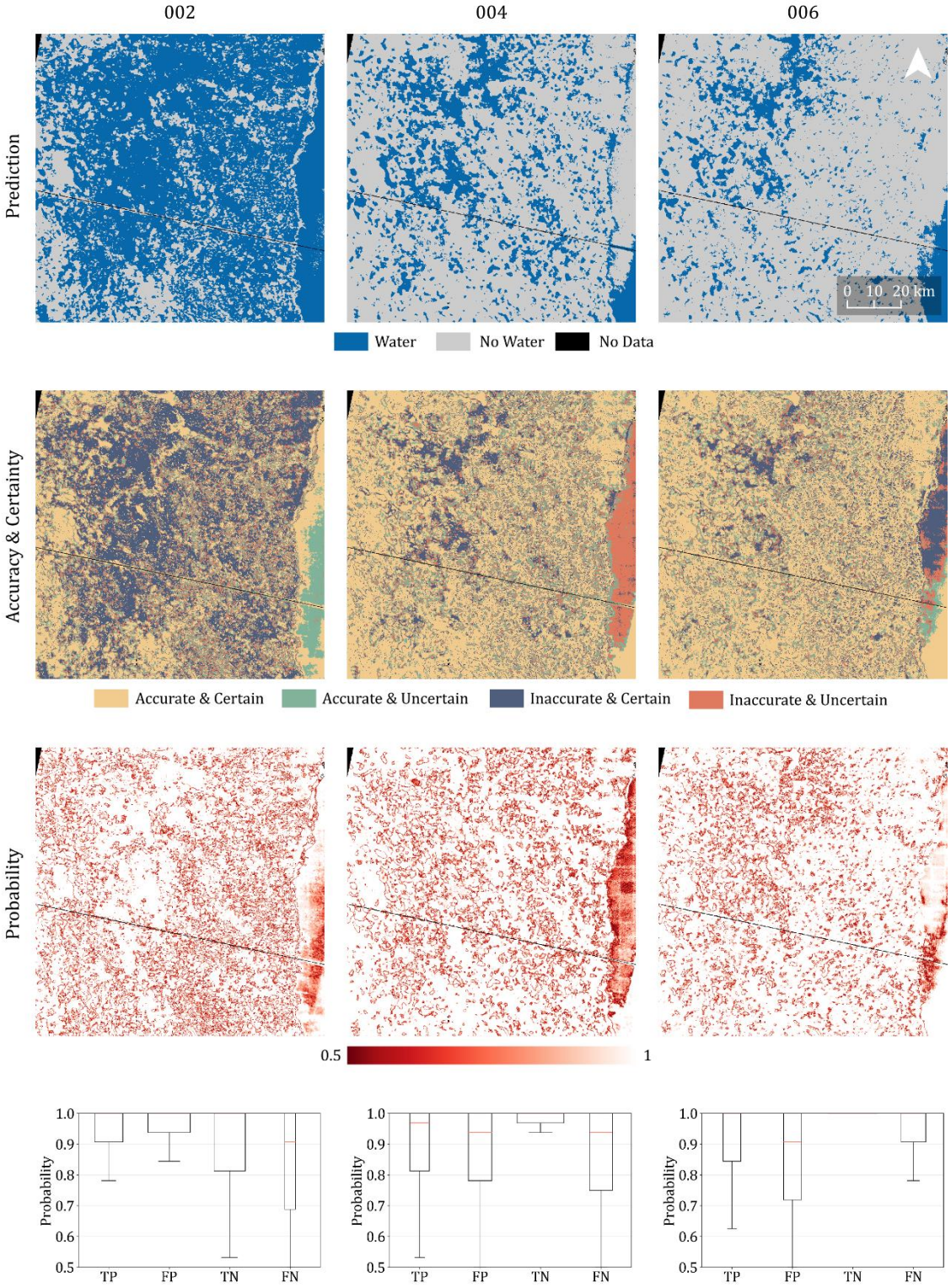
Scene 29, UD2:


002
004
006
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Probability
0.5
1
Probability
1.0
0.9
0.8
0.7
0.6
0.5
TP
FP
TN
FN

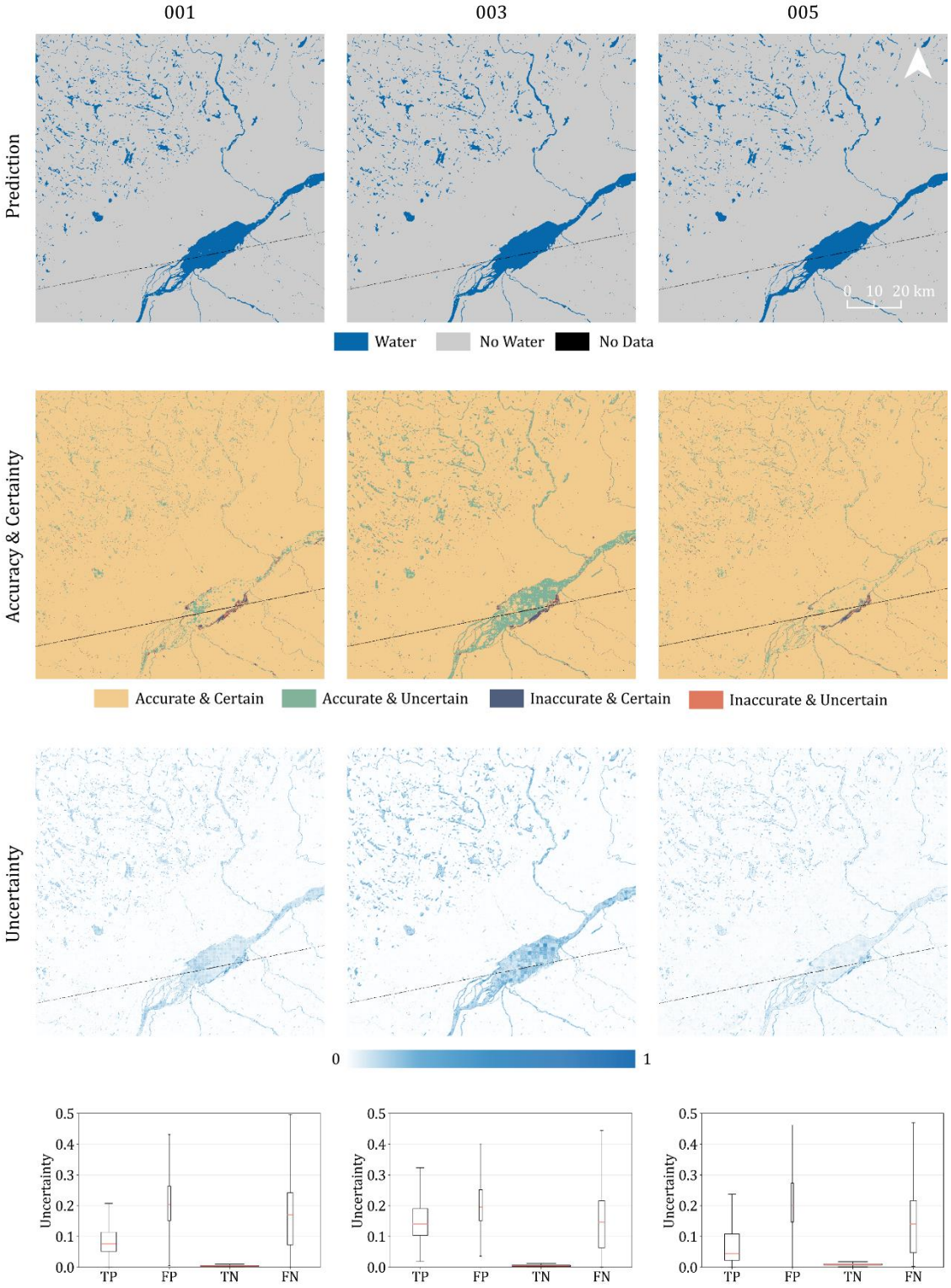
Scene 31, UD1:


001
003
005
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Uncertainty
0
1
Uncertainty
0.0
0.1
0.2
0.3
0.4
0.5
TP
FP
TN
FN

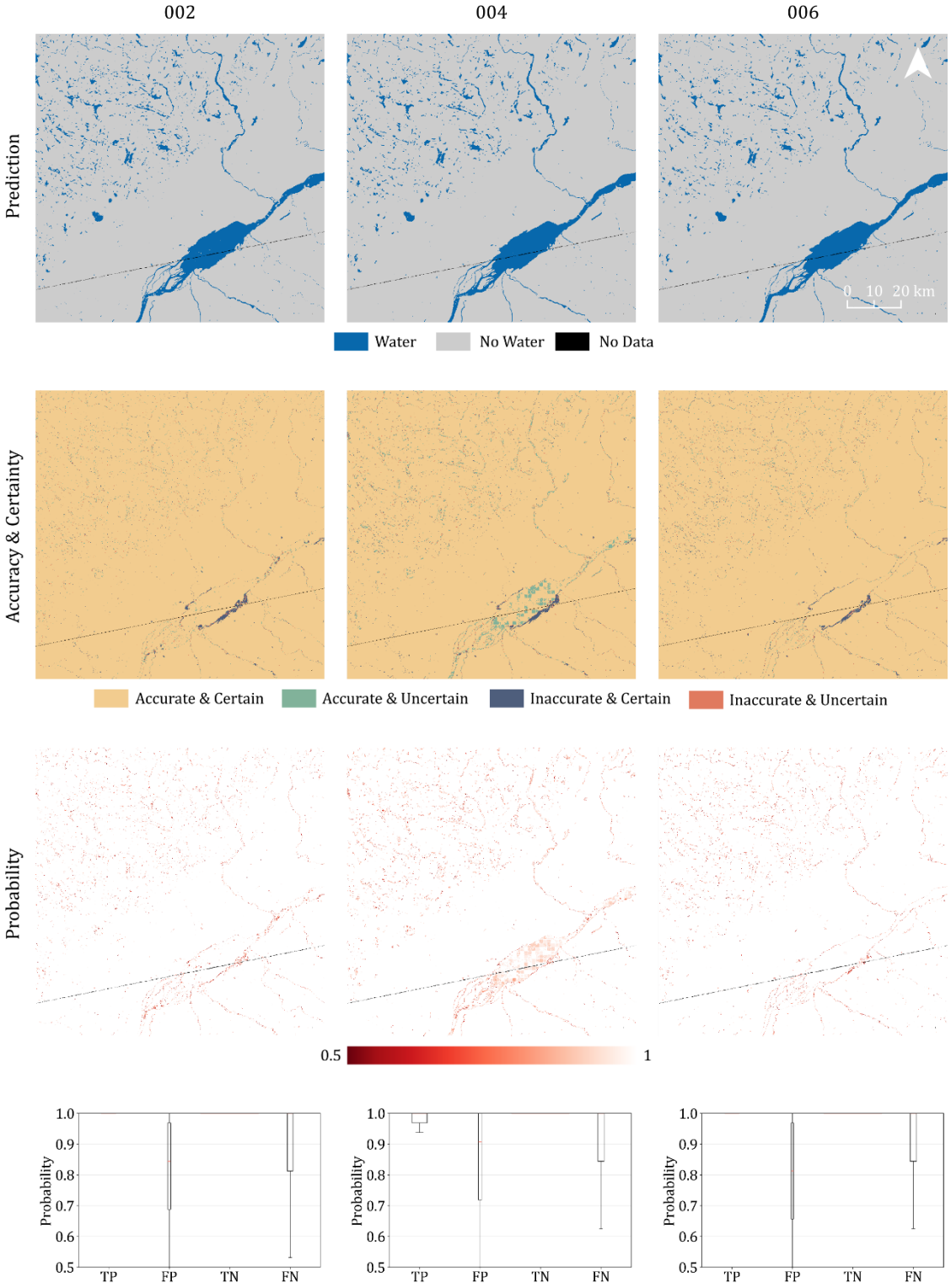
Scene 31, UD2:


002
004
006
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Probability
0.5
1
Probability
1.0
0.9
0.8
0.7
0.6
0.5
TP
FP
TN
FN

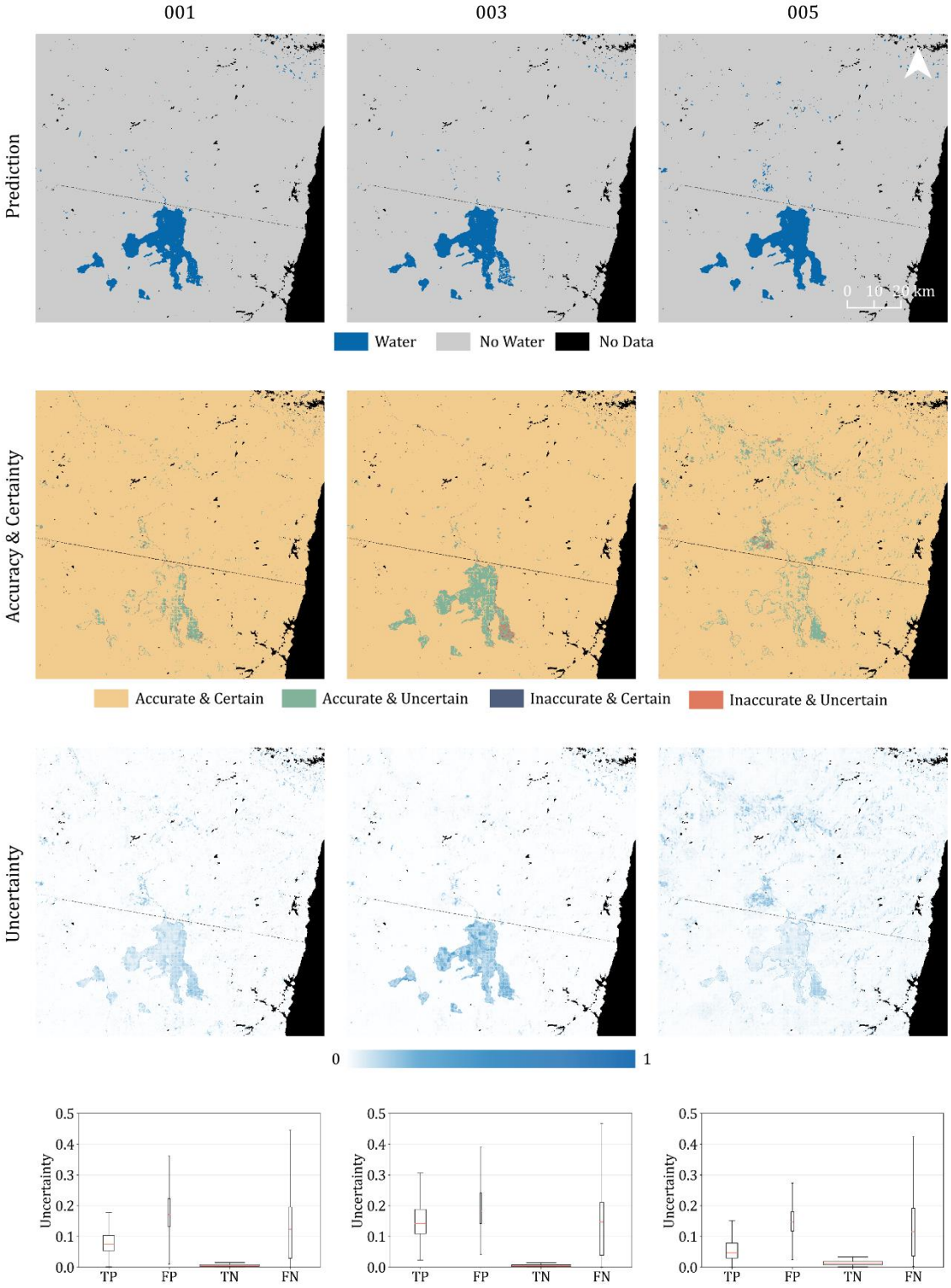
Scene 33, UD1:


001
003
005
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Uncertainty
0
1
Uncertainty
0.0
0.1
0.2
0.3
0.4
0.5
TP
FP
TN
FN

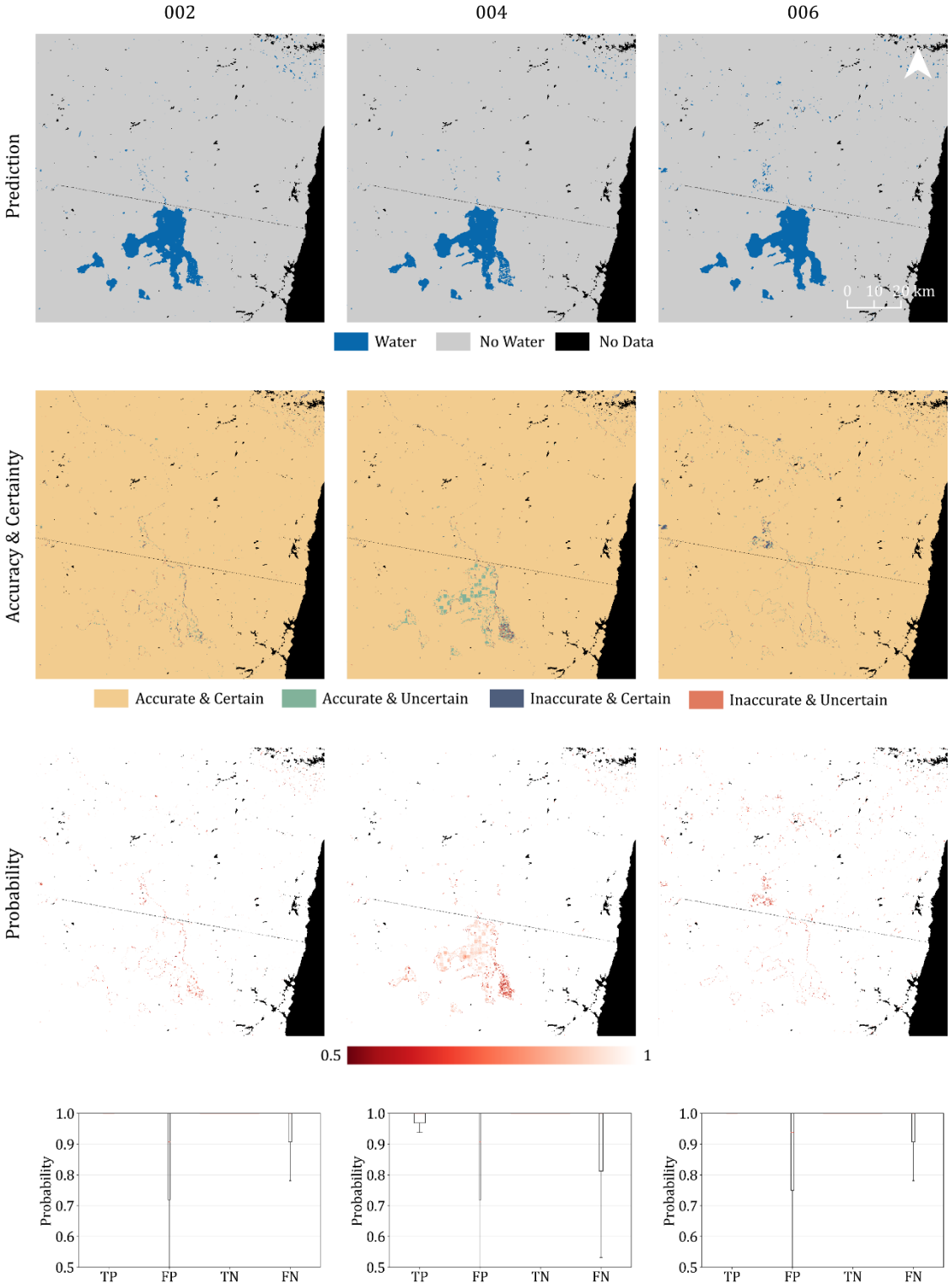
Scene 33, UD2:


002
004
006
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Probability
0.5
1
Probability
1.0
0.9
0.8
0.7
0.6
0.5
TP
FP
TN
FN

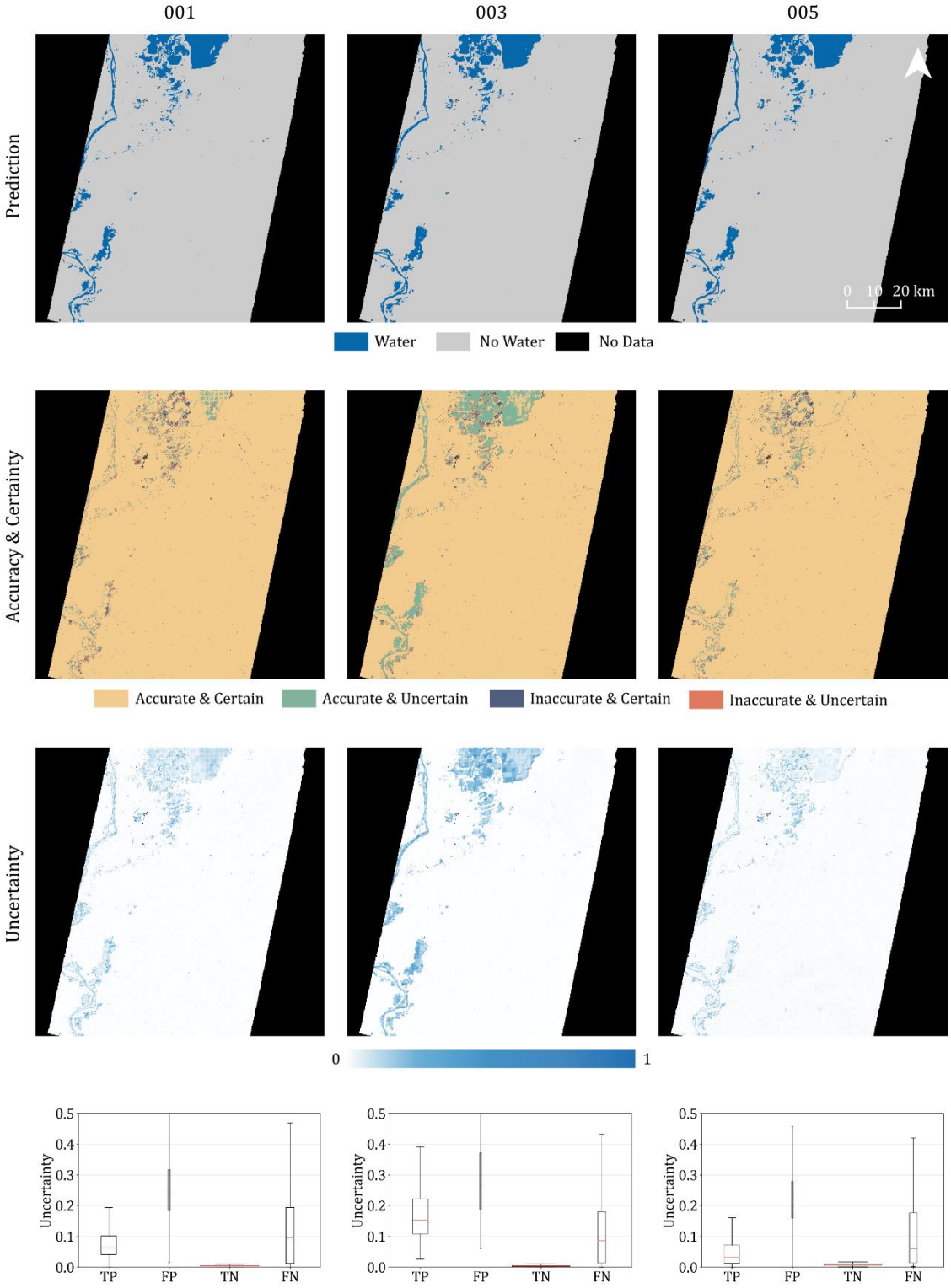
Scene 35, UD1:


001
003
005
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Uncertainty
0
1
Uncertainty
0.5
0.4
0.3
0.2
0.1
0.0
TP
FP
TN
FN

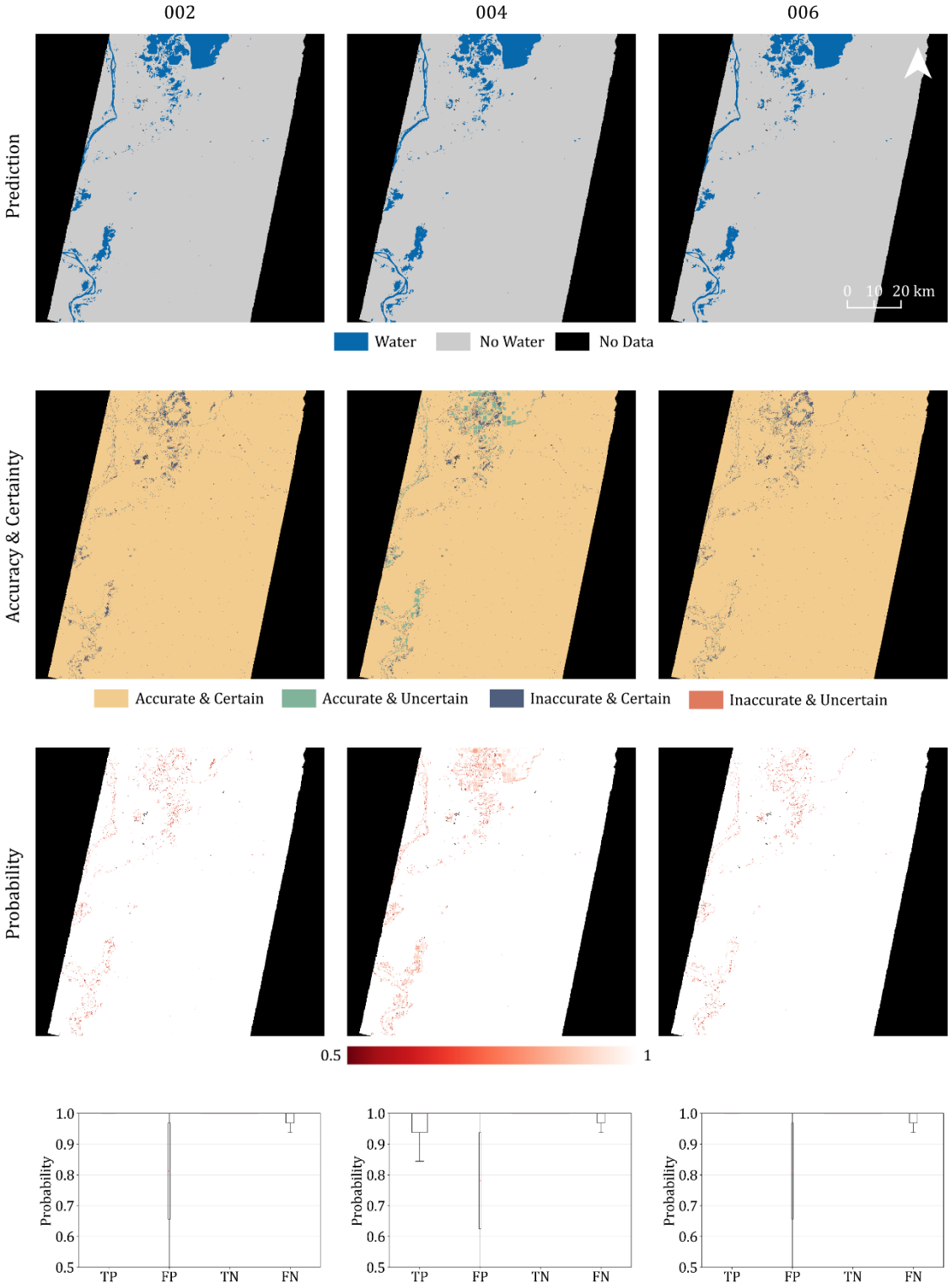
Scene 35, UD2:


002
004
006
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Probability
0.5
1
Probability
1.0
0.9
0.8
0.7
0.6
0.5
TP
FP
TN
FN

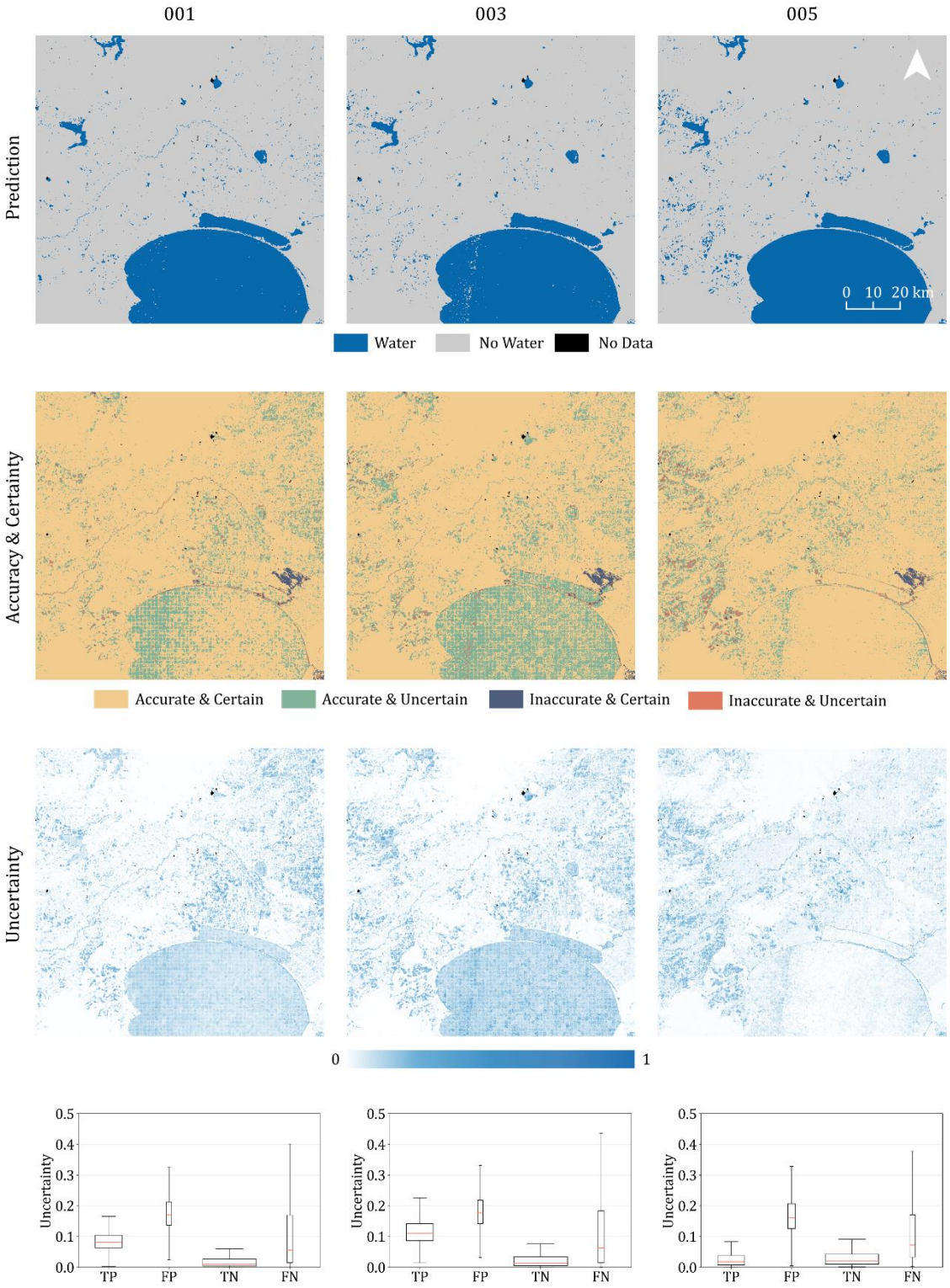
Scene 47, UD1:


001
003
005
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Uncertainty
0
1
Uncertainty
0.5
0.4
0.3
0.2
0.1
0.0
TP
FP
TN
FN

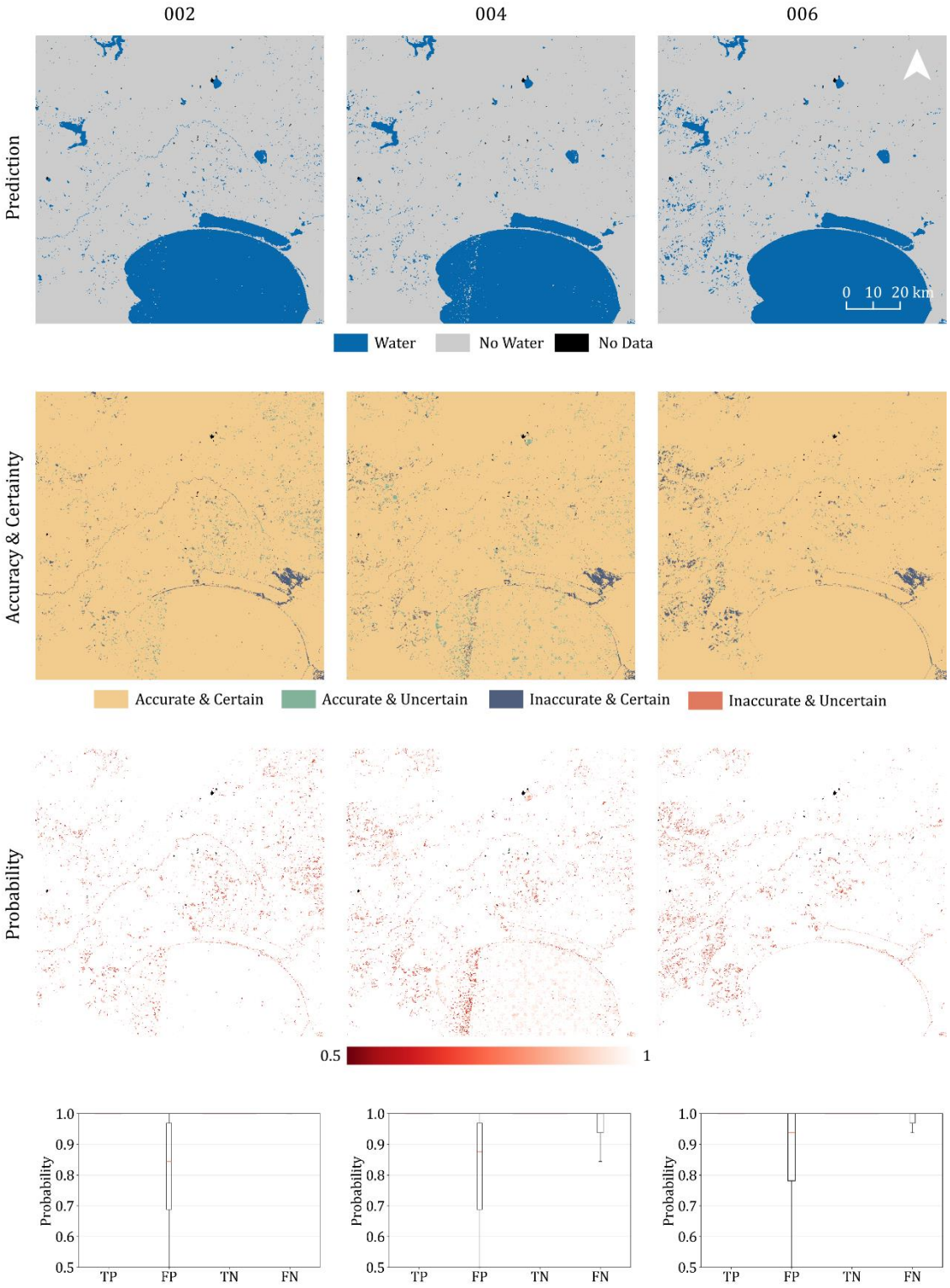
Scene 47, UD2:


002
004
006
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Probability
0.5
1
Probability
1.0
0.9
0.8
0.7
0.6
0.5
TP
FP
TN
FN

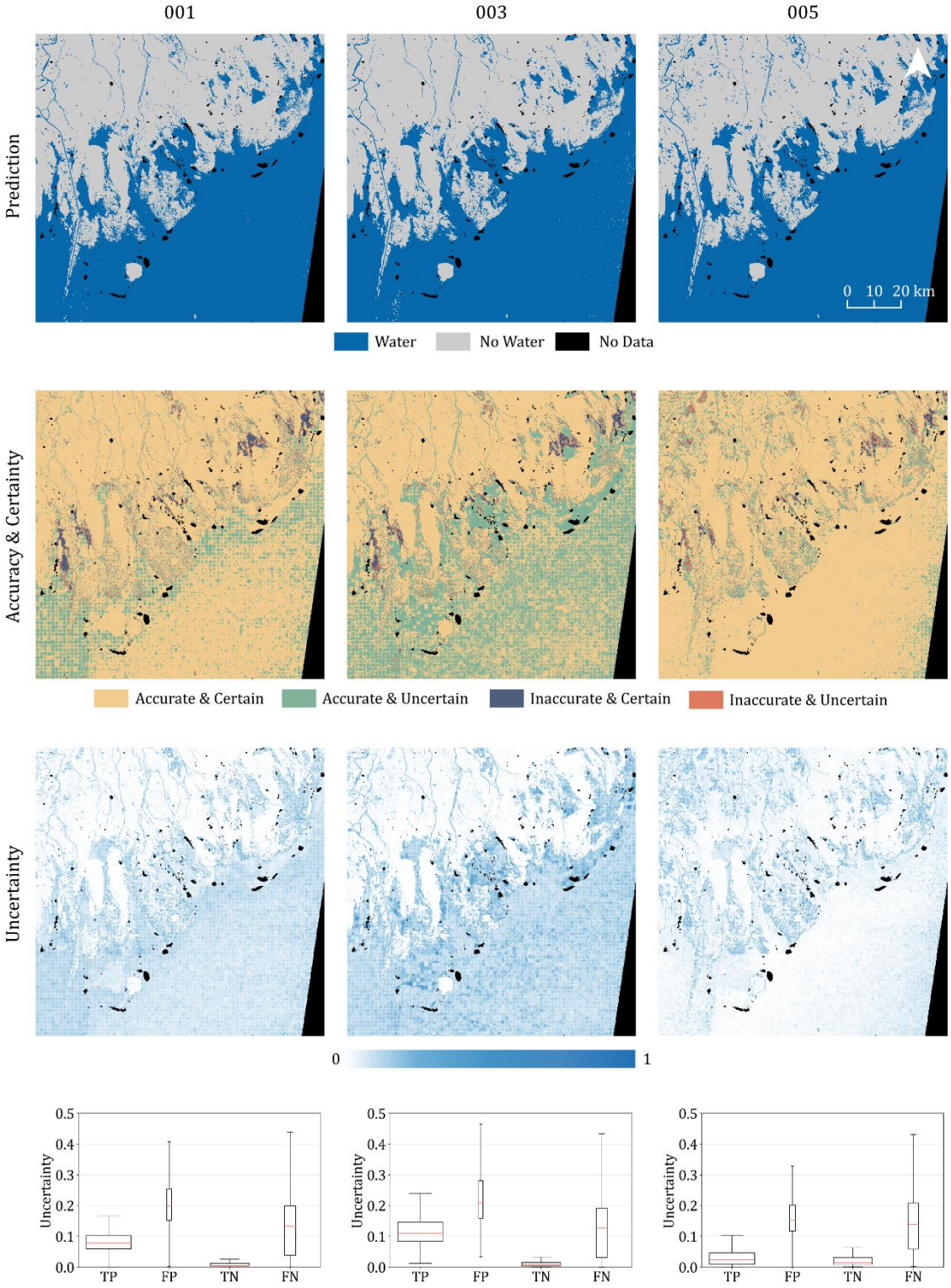
Scene 53, UD1:


001
003
005
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Uncertainty
0
1
Uncertainty
0.5
0.4
0.3
0.2
0.1
0.0
TP
FP
TN
FN

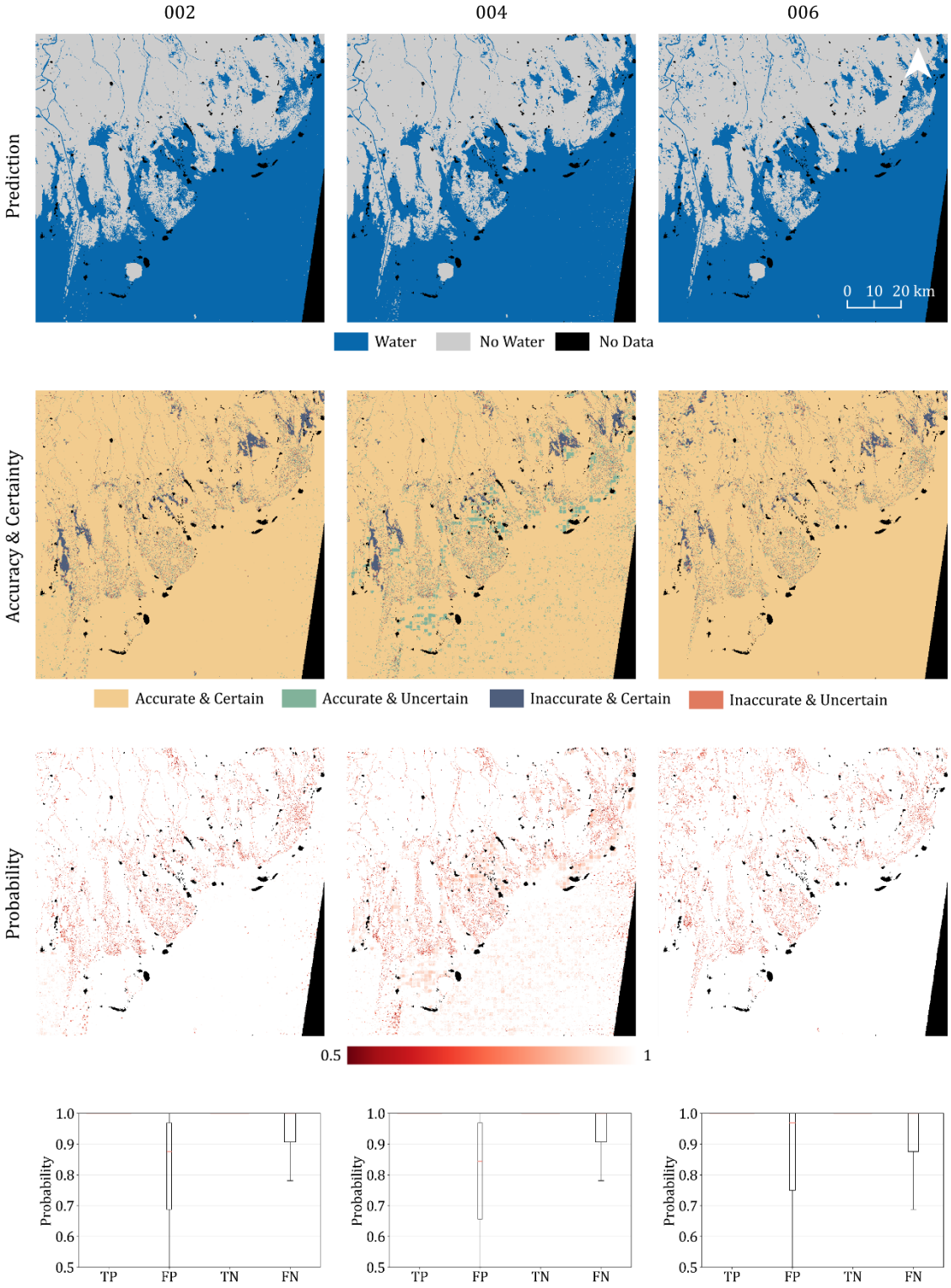
Scene 53, UD2:


002
004
006
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Probability
0.5
1
Probability
1.0
0.9
0.8
0.7
0.6
0.5
TP
FP
TN
FN

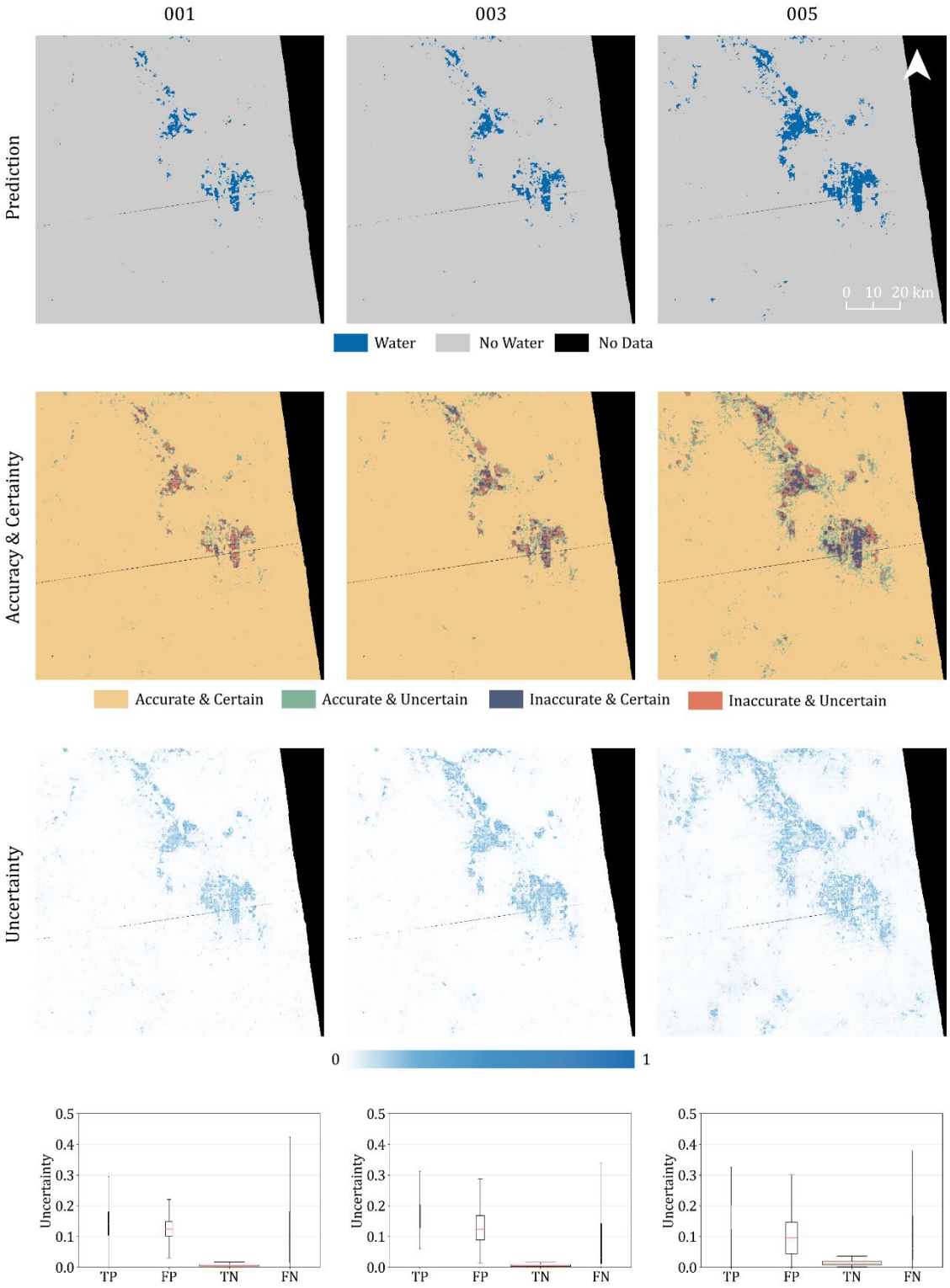
Scene 57, UD1:


001
003
005
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Uncertainty
0
1
Uncertainty
0.0
0.1
0.2
0.3
0.4
0.5
TP
FP
TN
FN

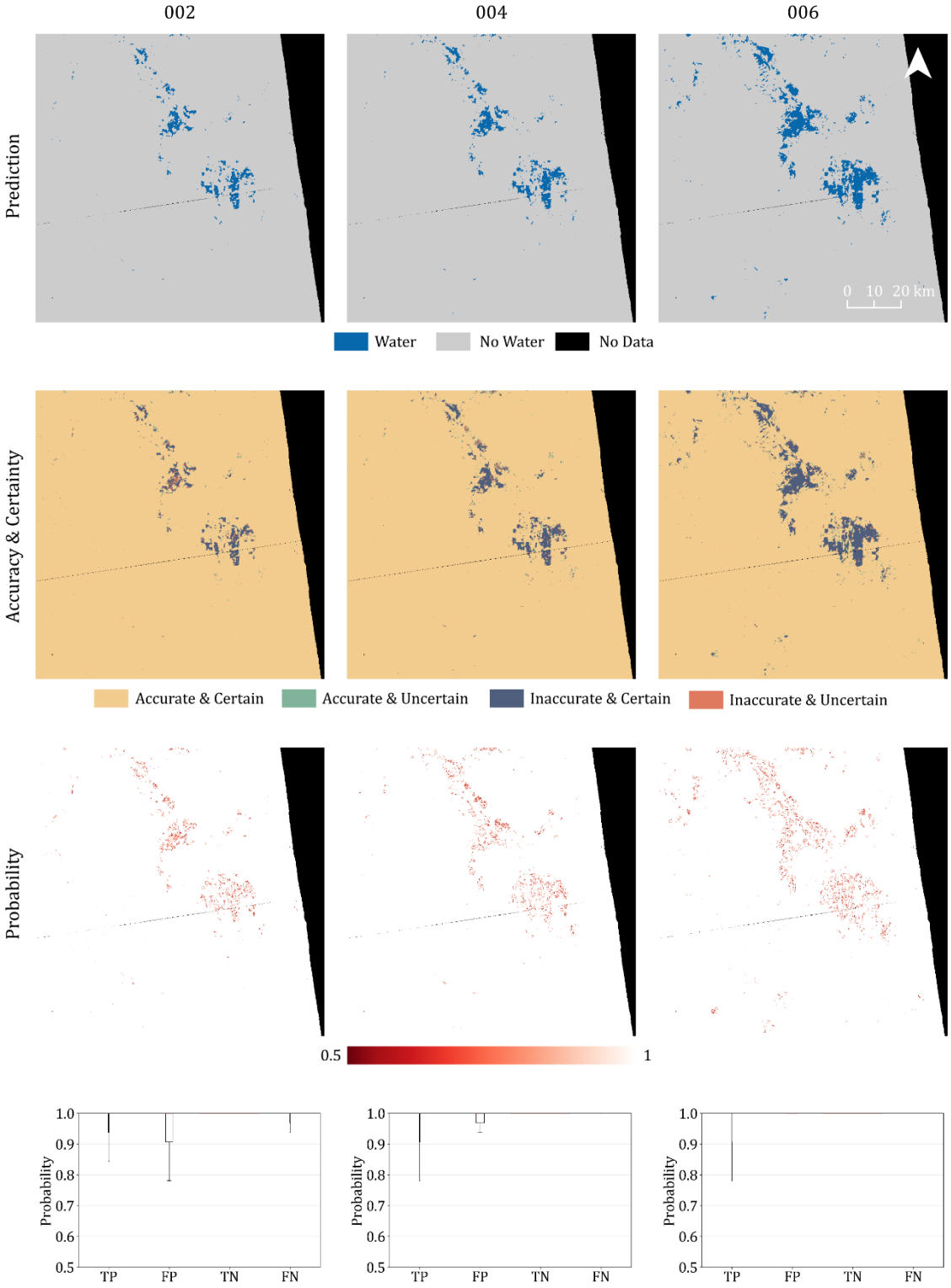
Scene 57, UD2:


002
004
006
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Probability
0.5
1
Probability
1.0
0.9
0.8
0.7
0.6
0.5
TP
FP
TN
FN

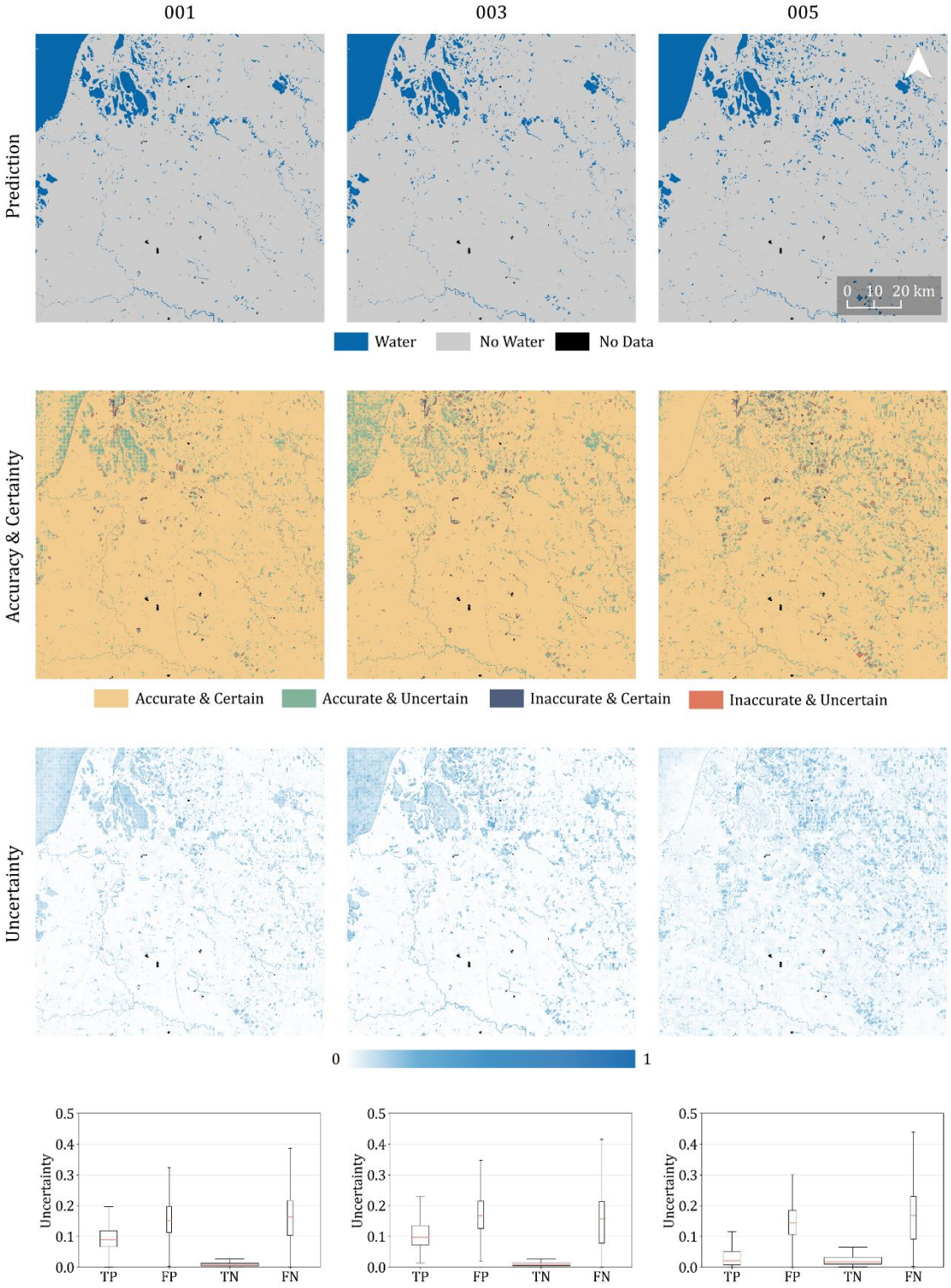
Scene 66, UD1:


001
003
005
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Uncertainty
0
1
Uncertainty
0.5
0.4
0.3
0.2
0.1
0.0
TP
FP
TN
FN

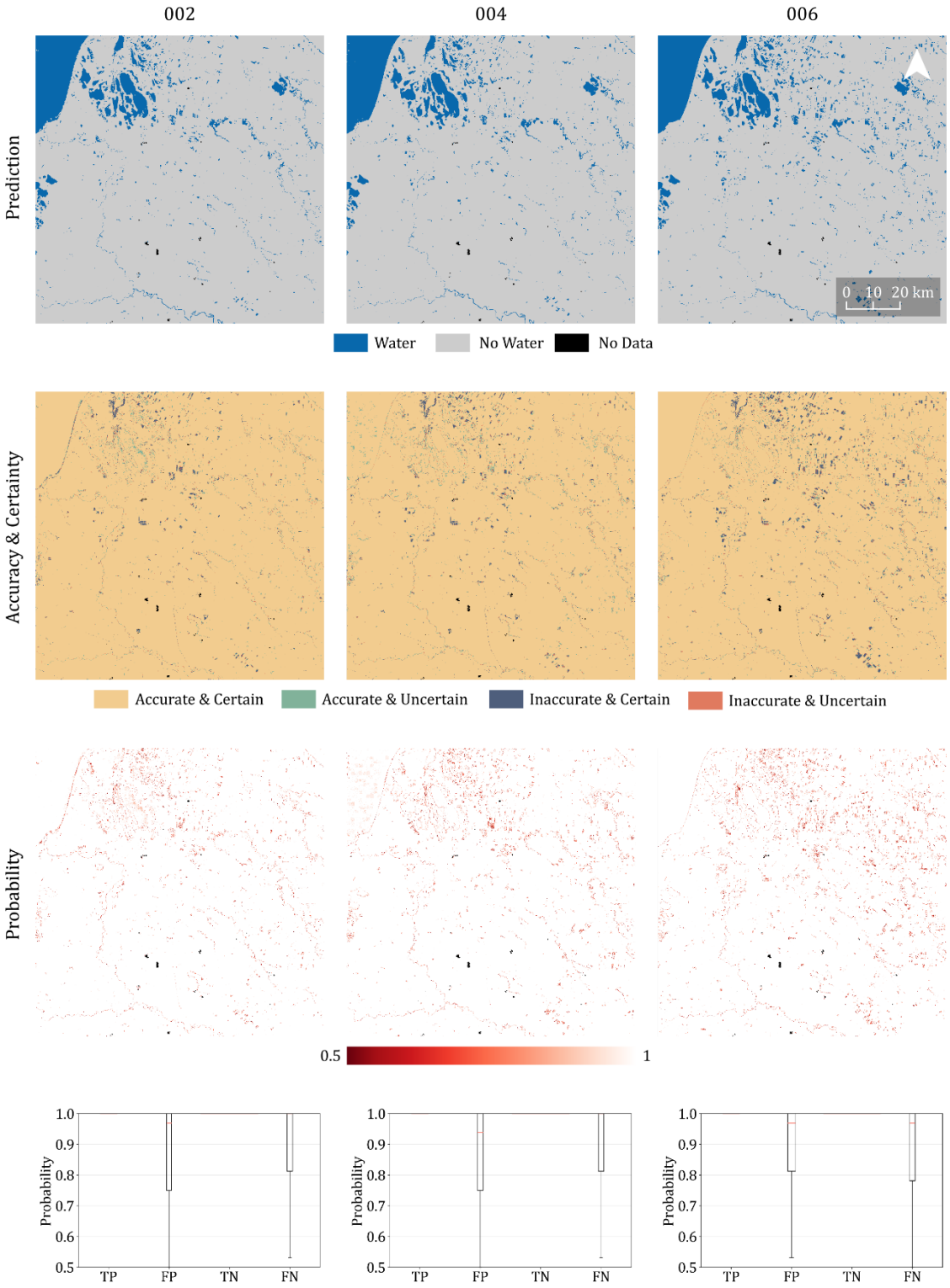
Scene 66, UD2:


002
004
006
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Probability
0.5
1
Probability
1.0
0.9
0.8
0.7
0.6
0.5
TP
FP
TN
FN

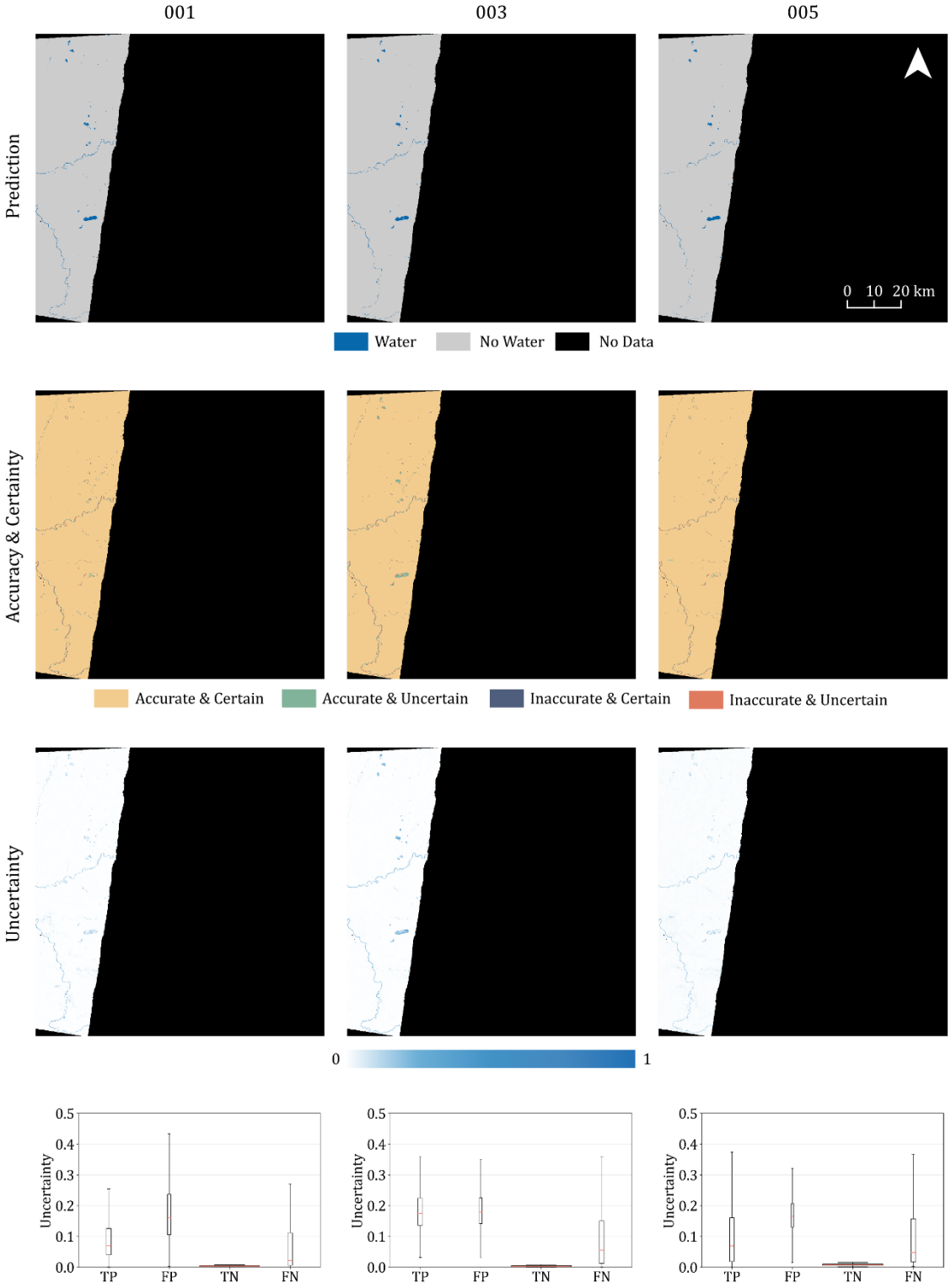
Scene 75, UD1:


001
003
005
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Uncertainty
0
1
Uncertainty
0.5
0.4
0.3
0.2
0.1
0.0
TP
FP
TN
FN

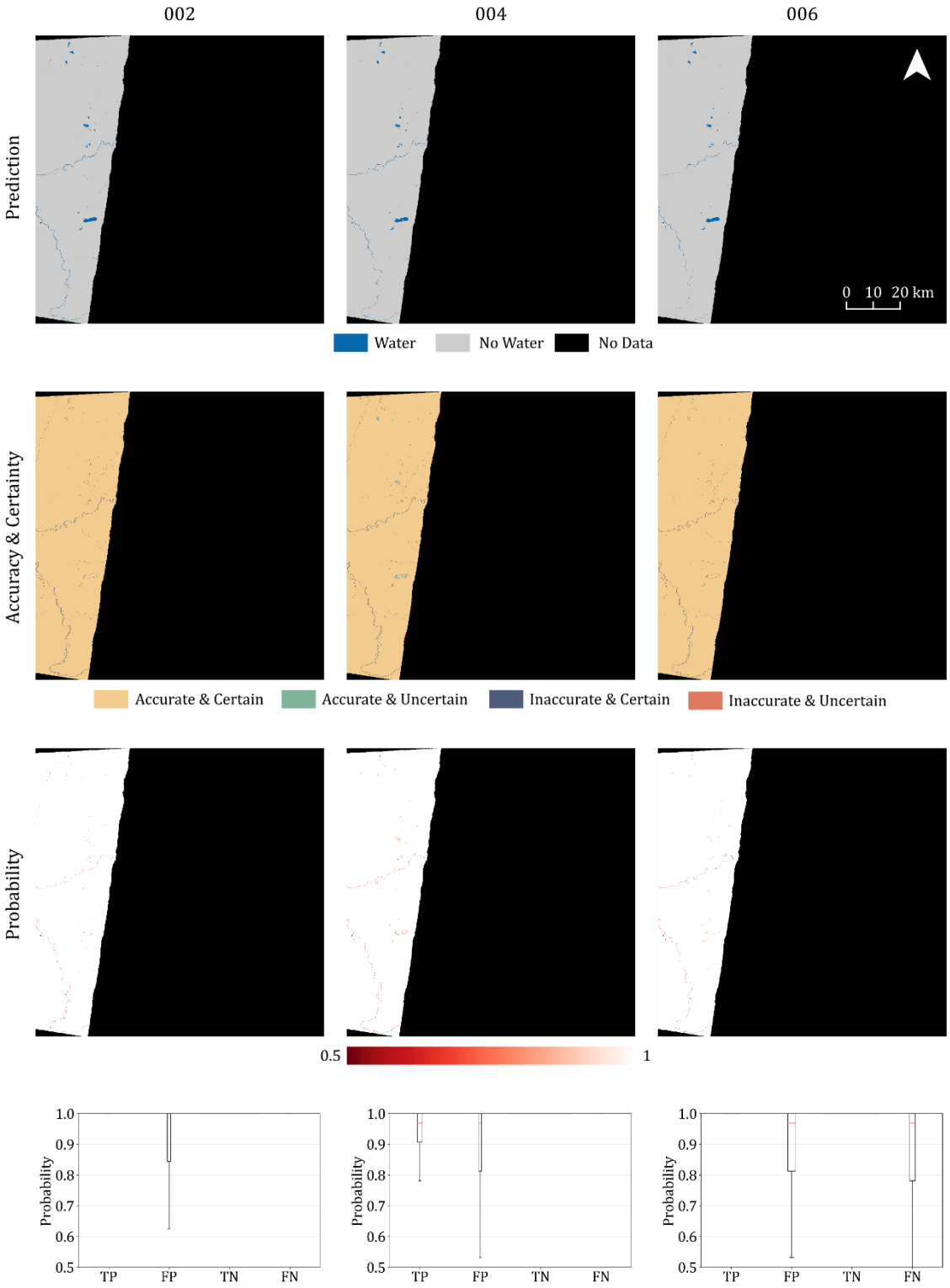
Scene 75, UD2:


002
004
006
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Probability
0.5
1
Probability
1.0
0.9
0.8
0.7
0.6
0.5
TP
FP
TN
FN

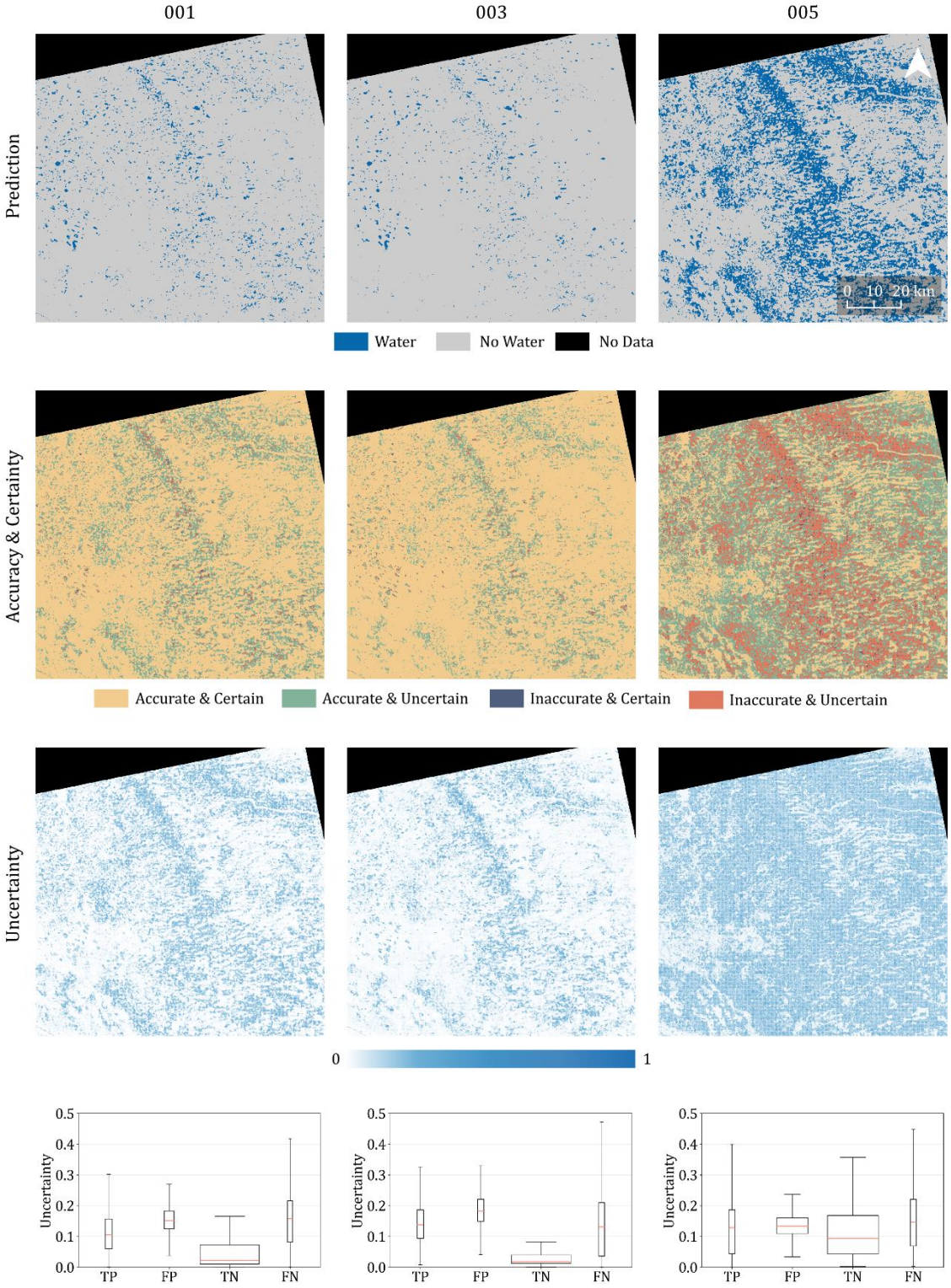
Scene 77, UD1:


001
003
005
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Uncertainty
0
1
Uncertainty
0.5
0.4
0.3
0.2
0.1
0.0
TP
FP
TN
FN

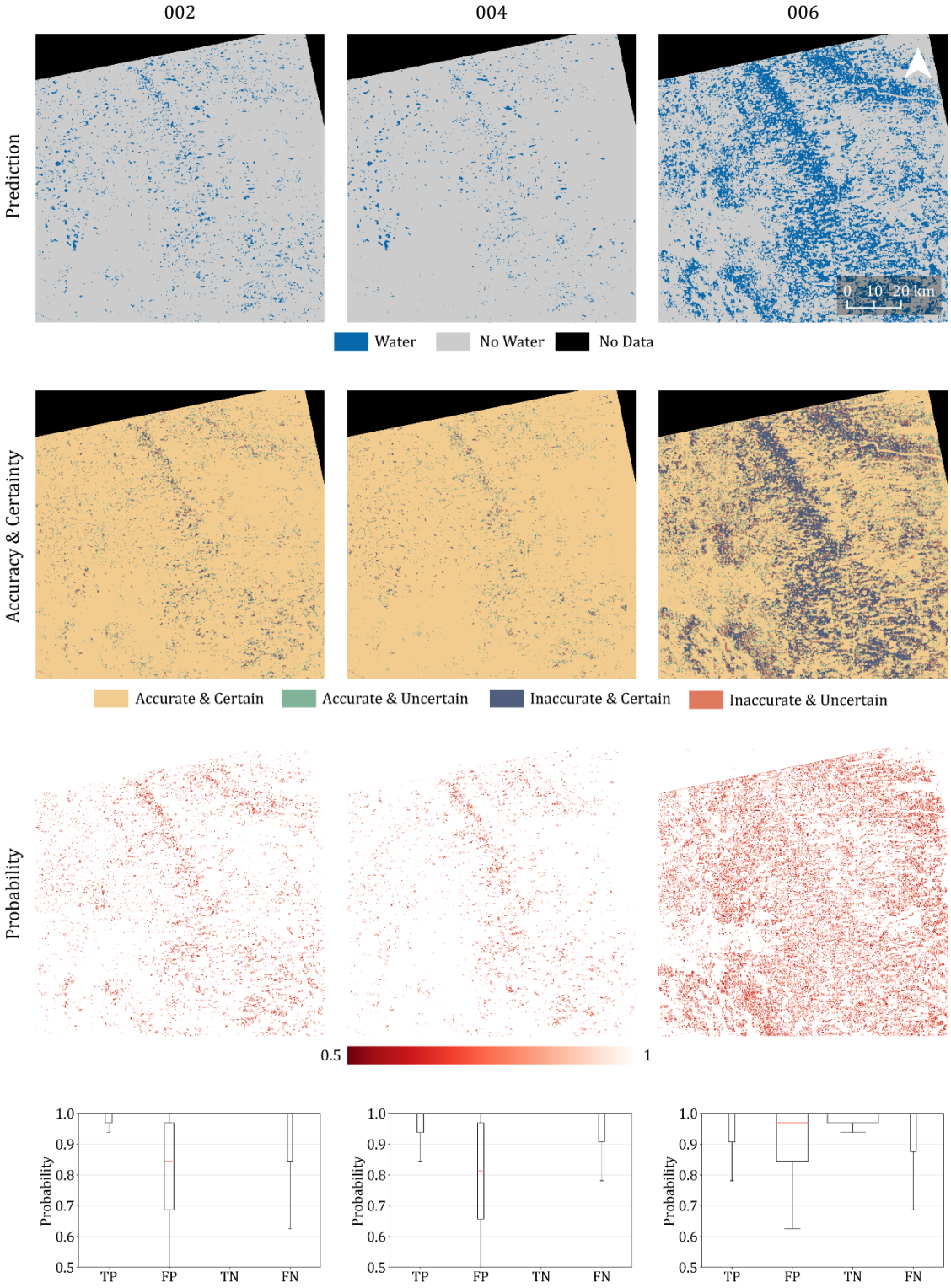
Scene 77, UD2:


002
004
006
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Probability
0.5
1
Probability
1.0
0.9
0.8
0.7
0.6
0.5
TP
FP
TN
FN

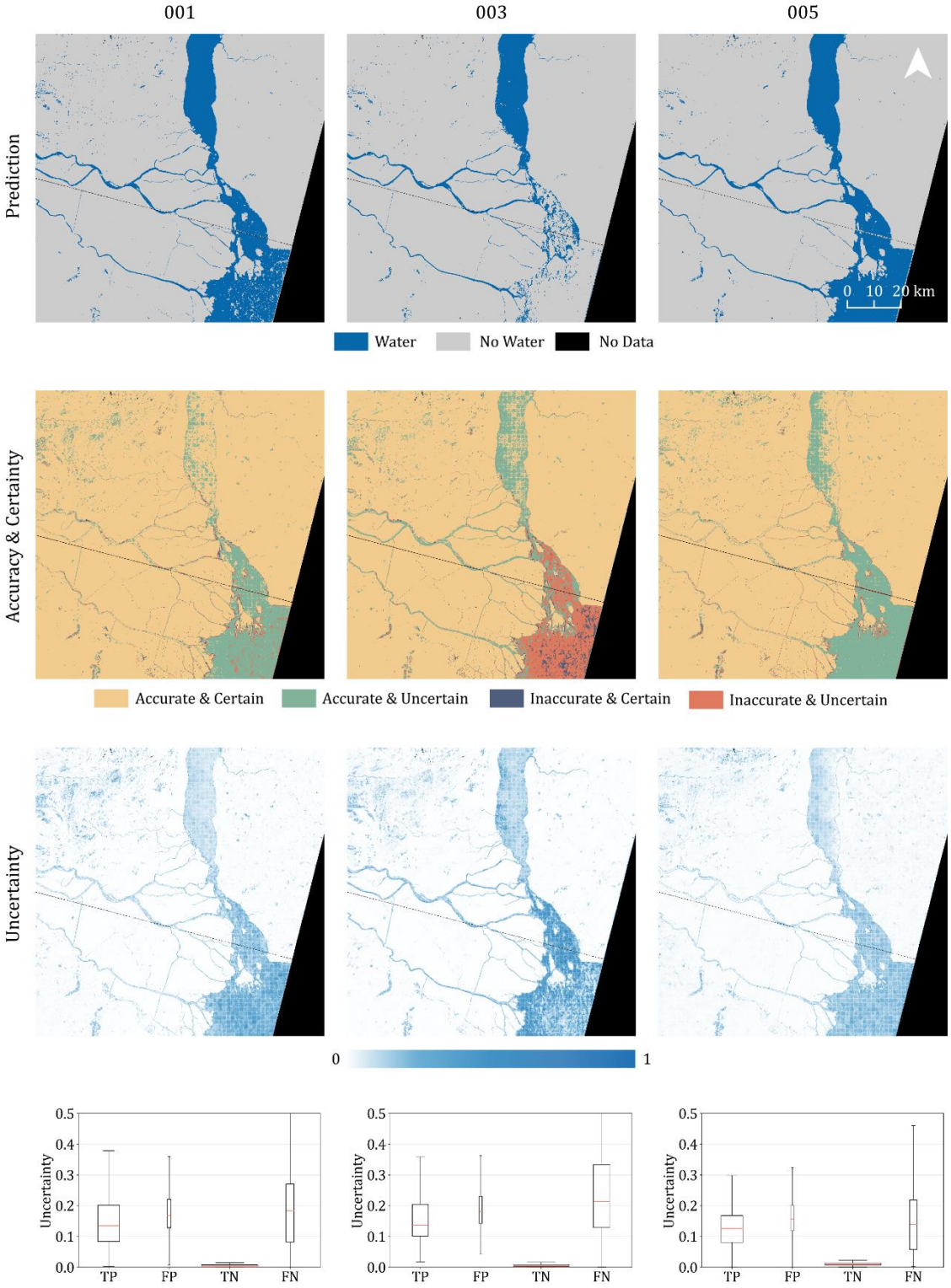
Scene 78, UD1:


001
003
005
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Uncertainty
0
1
Uncertainty
0.0
0.1
0.2
0.3
0.4
0.5
TP
FP
TN
FN

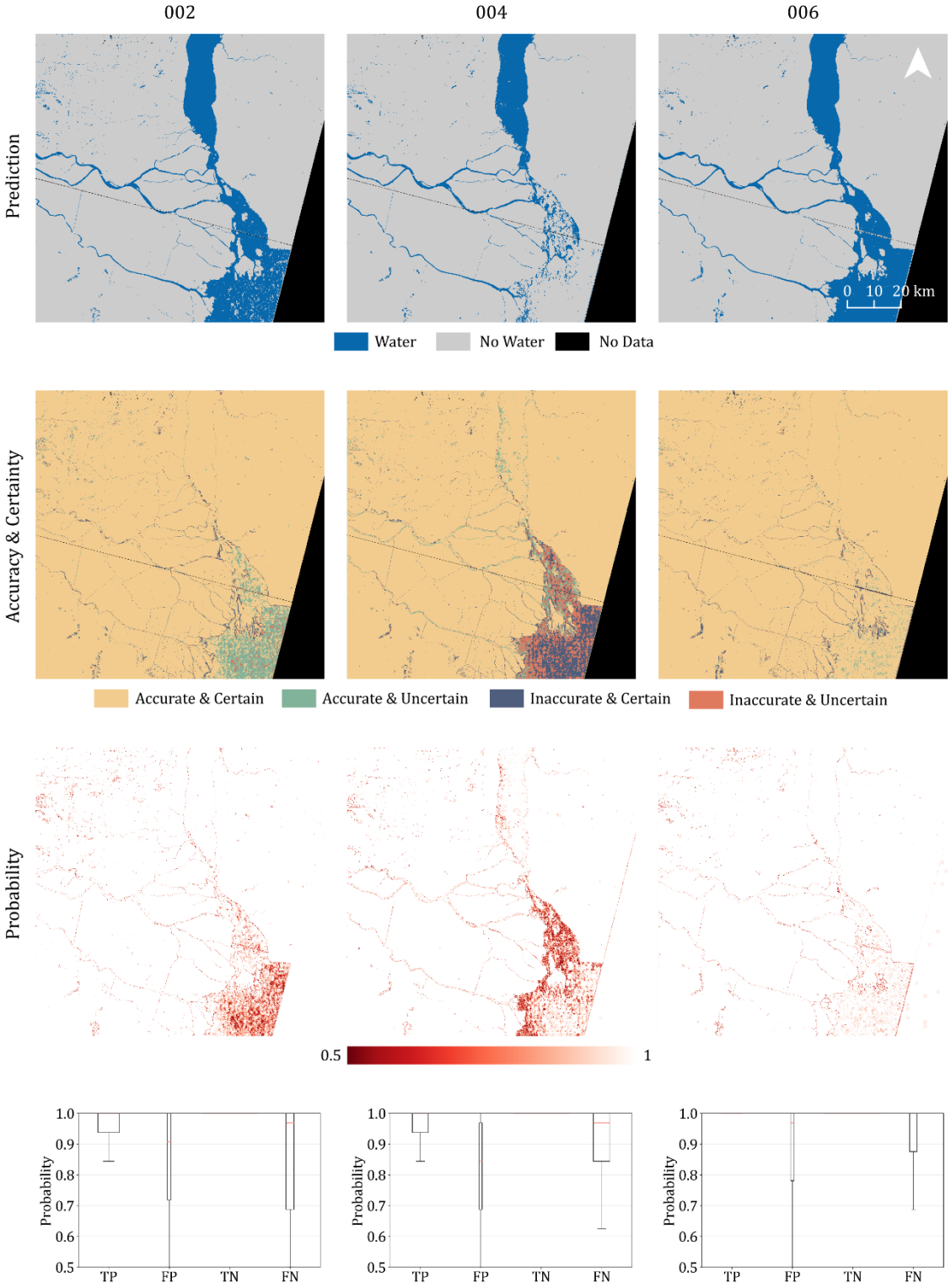
Scene 78, UD2:


002
004
006
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Probability
0.5
1
Probability
1.0
0.9
0.8
0.7
0.6
0.5
TP
FP
TN
FN

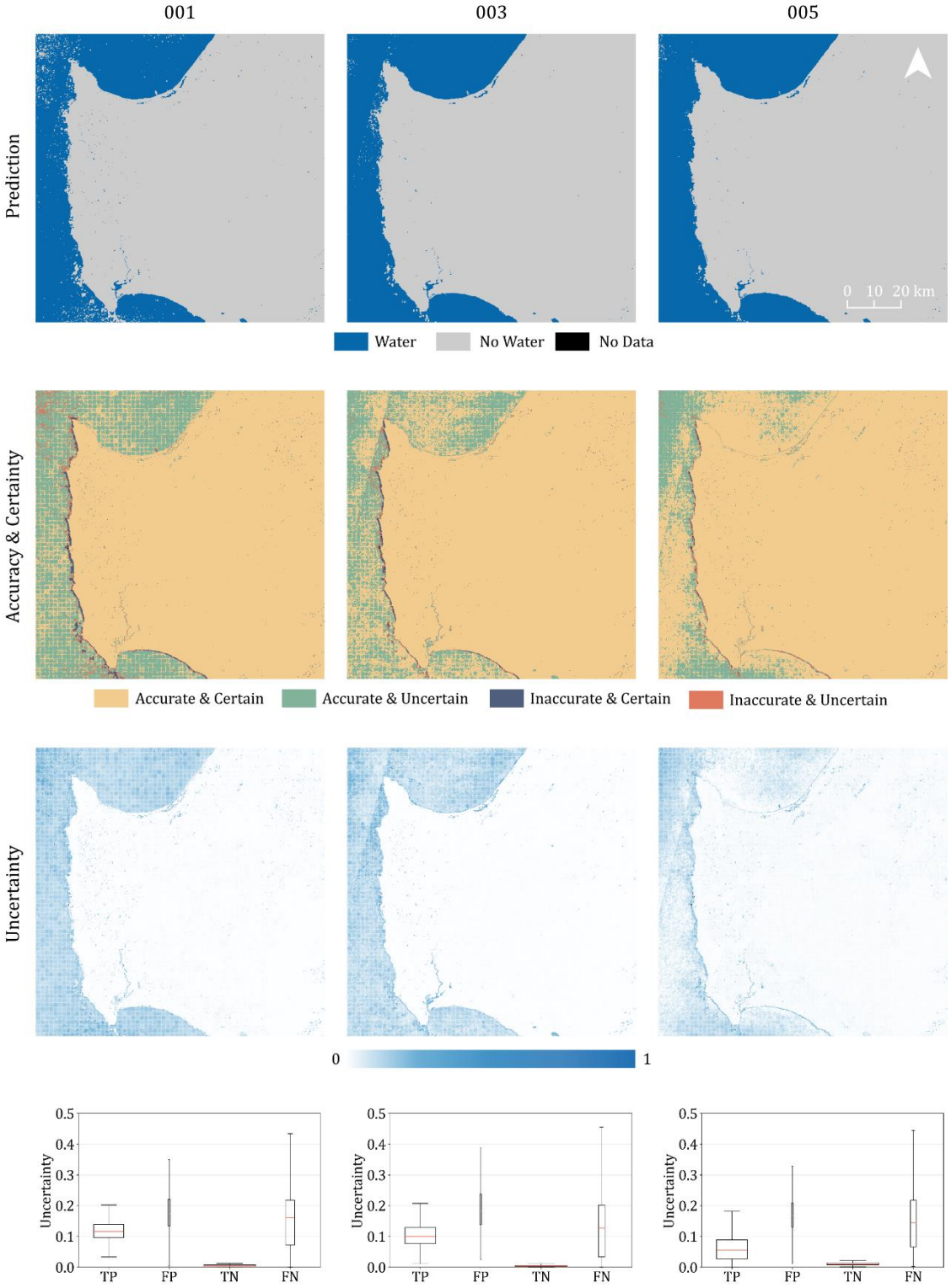
Scene 80, UD1:


001
003
005
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Uncertainty
0
1
Uncertainty
0.0
0.1
0.2
0.3
0.4
0.5
TP
FP
TN
FN

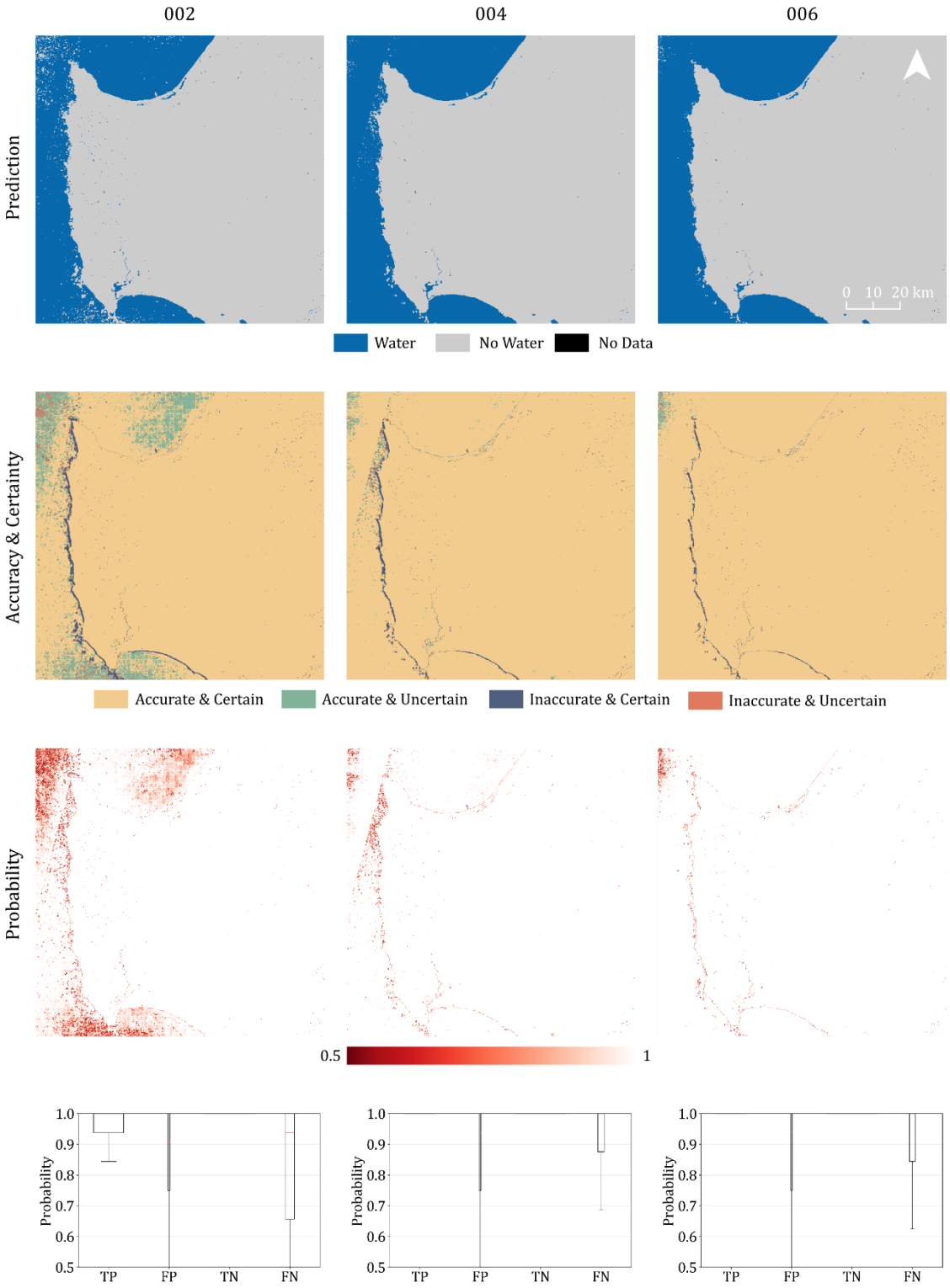
Scene 80, UD2:


002
004
006
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Probability
0.5
1
Probability
1.0
0.9
0.8
0.7
0.6
0.5
TP
FP
TN
FN

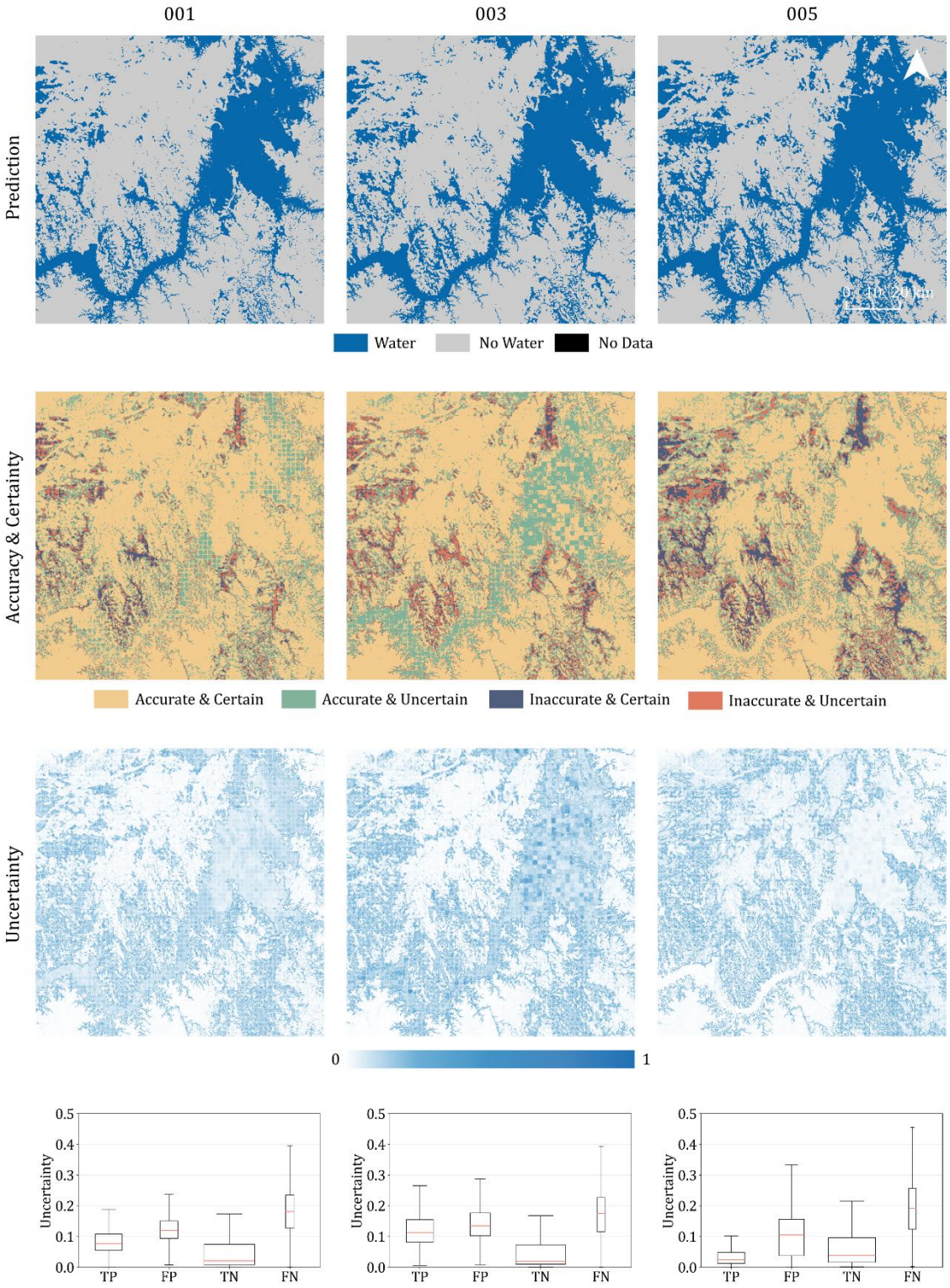
Scene 82, UD1:


001
003
005
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Uncertainty
0
1
Uncertainty
0.0
0.1
0.2
0.3
0.4
0.5
TP
FP
TN
FN

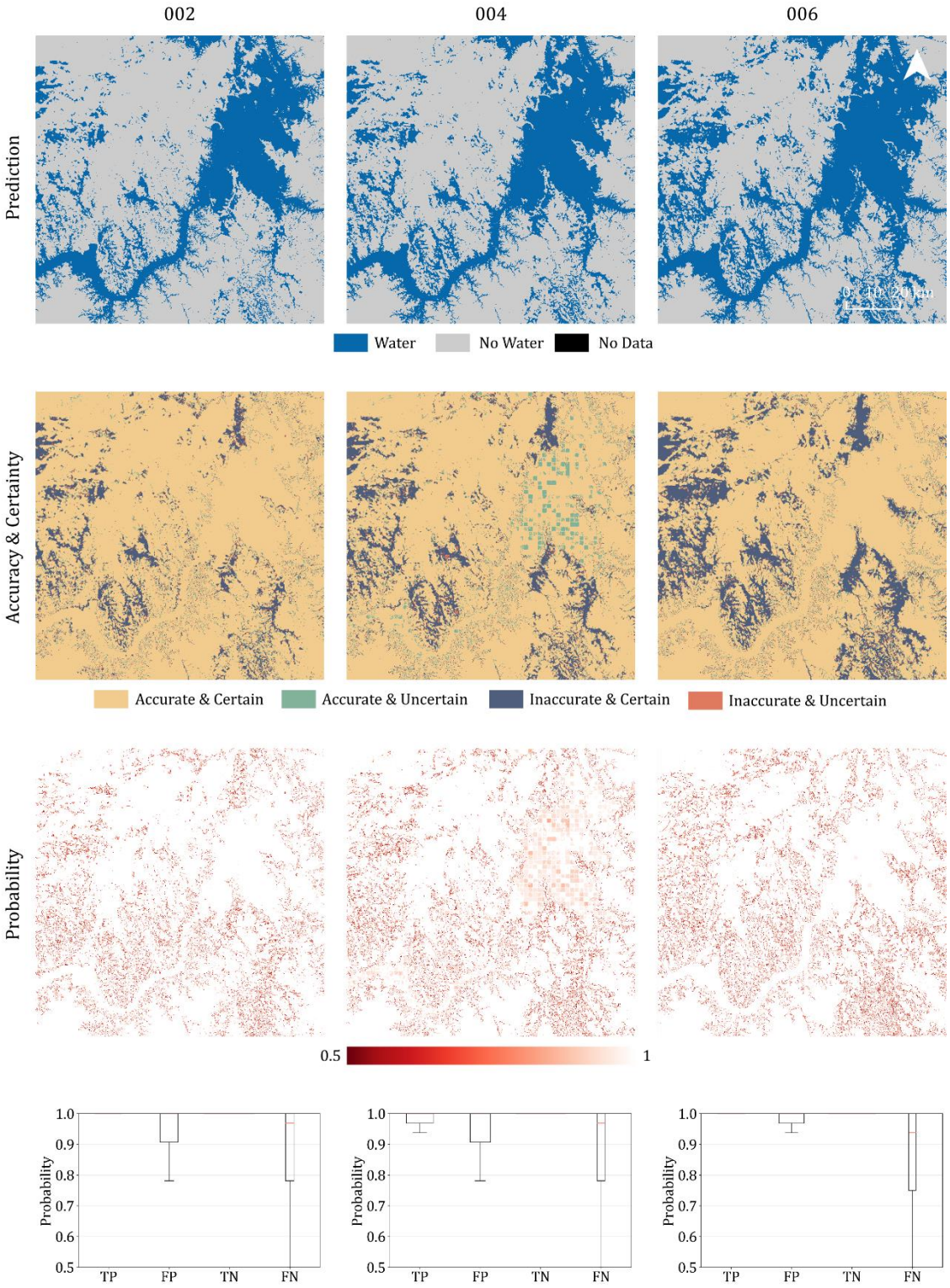
Scene 82, UD2:


002
004
006
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Probability
0.5
1
Probability
1.0
0.9
0.8
0.7
0.6
0.5
TP
FP
TN
FN

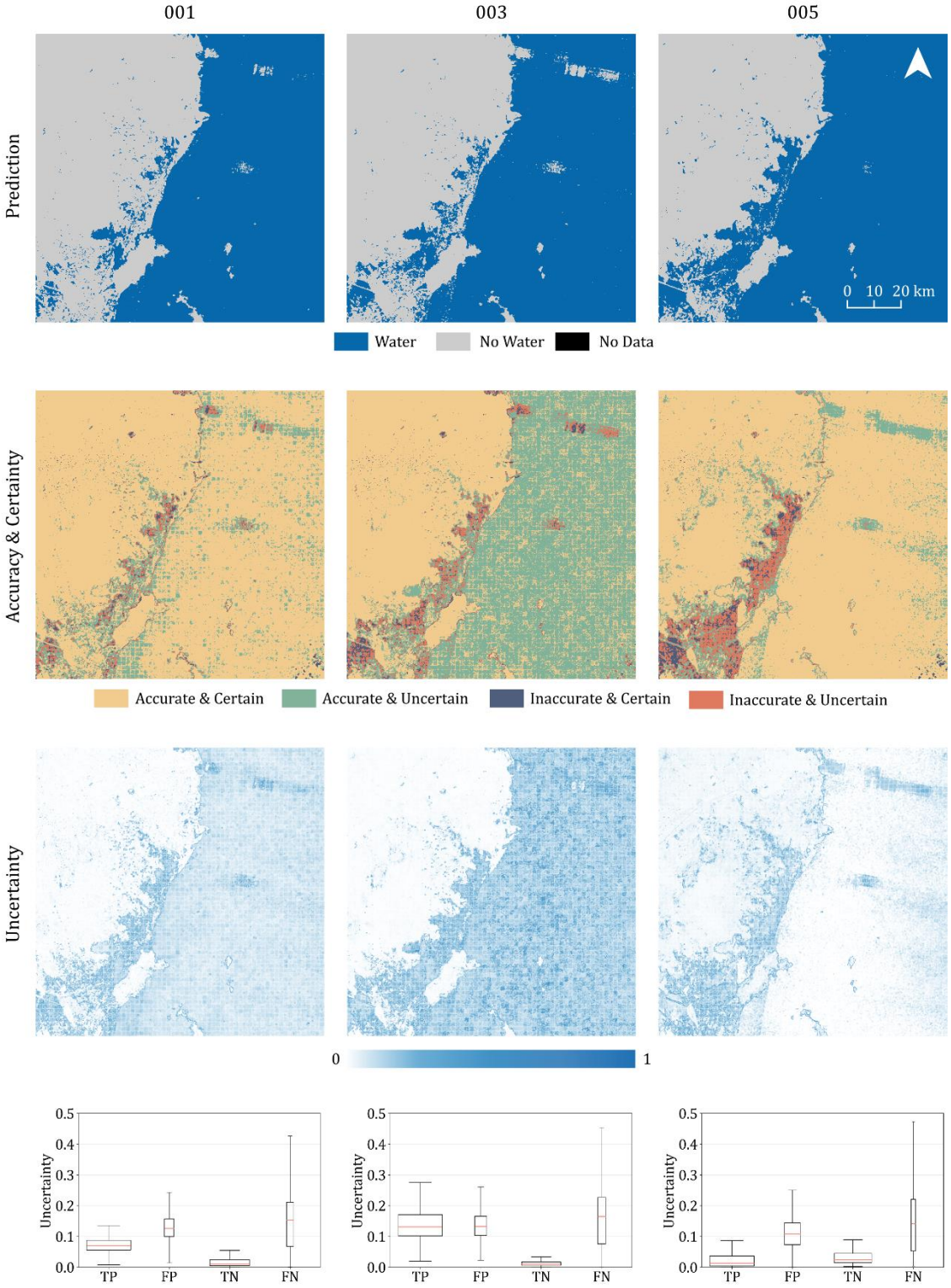
Scene 88, UD1:


001
003
005
Prediction
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Uncertainty
0
1
Uncertainty
0.5
0.4
0.3
0.2
0.1
0.0
TP
FP
TN
FN

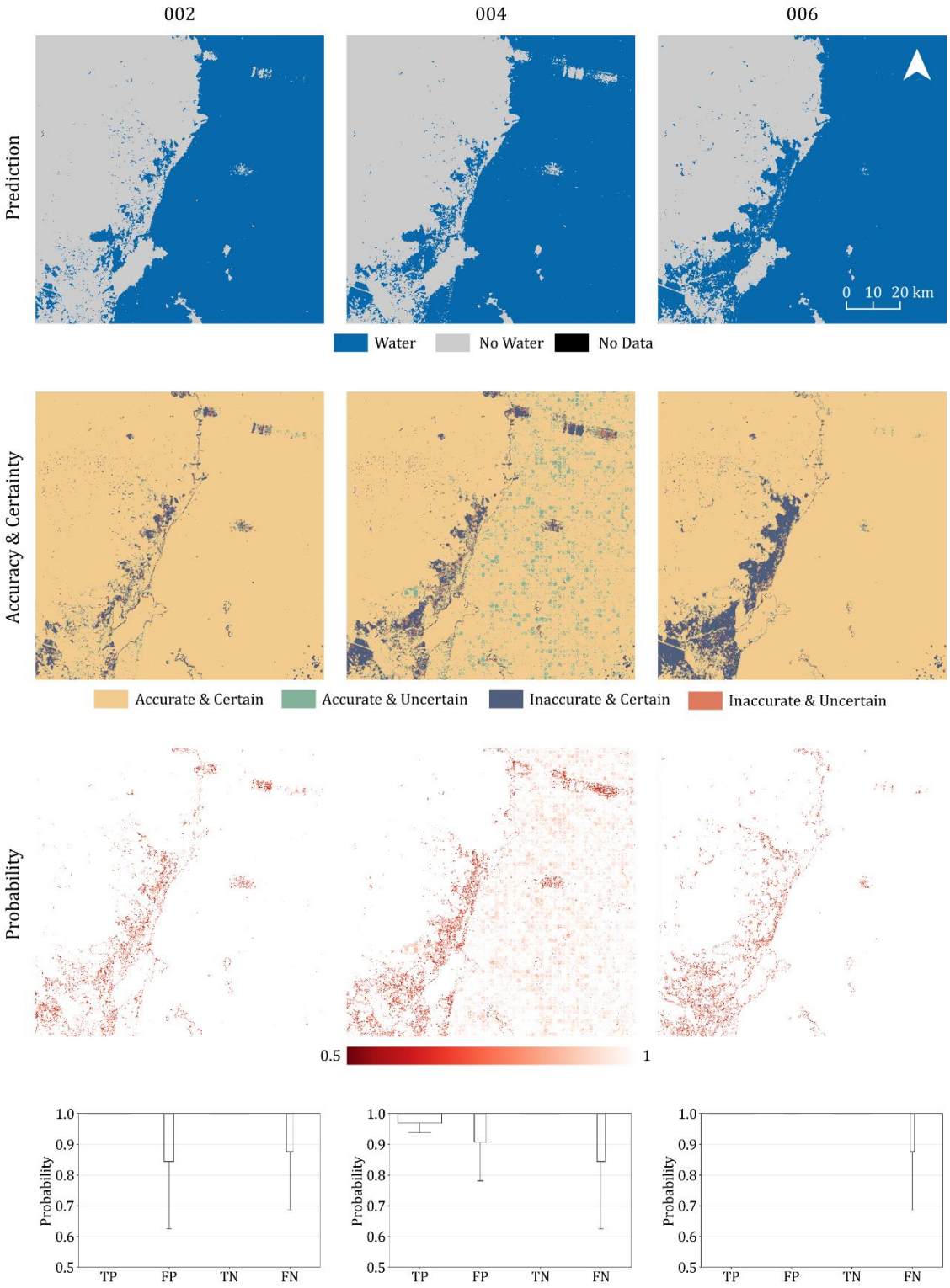
Scene 88, UD2:


002
004
006
Prediction
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Probability
0.5
1
Probability
1.0
0.9
0.8
0.7
0.6
0.5
TP
FP
TN
FN

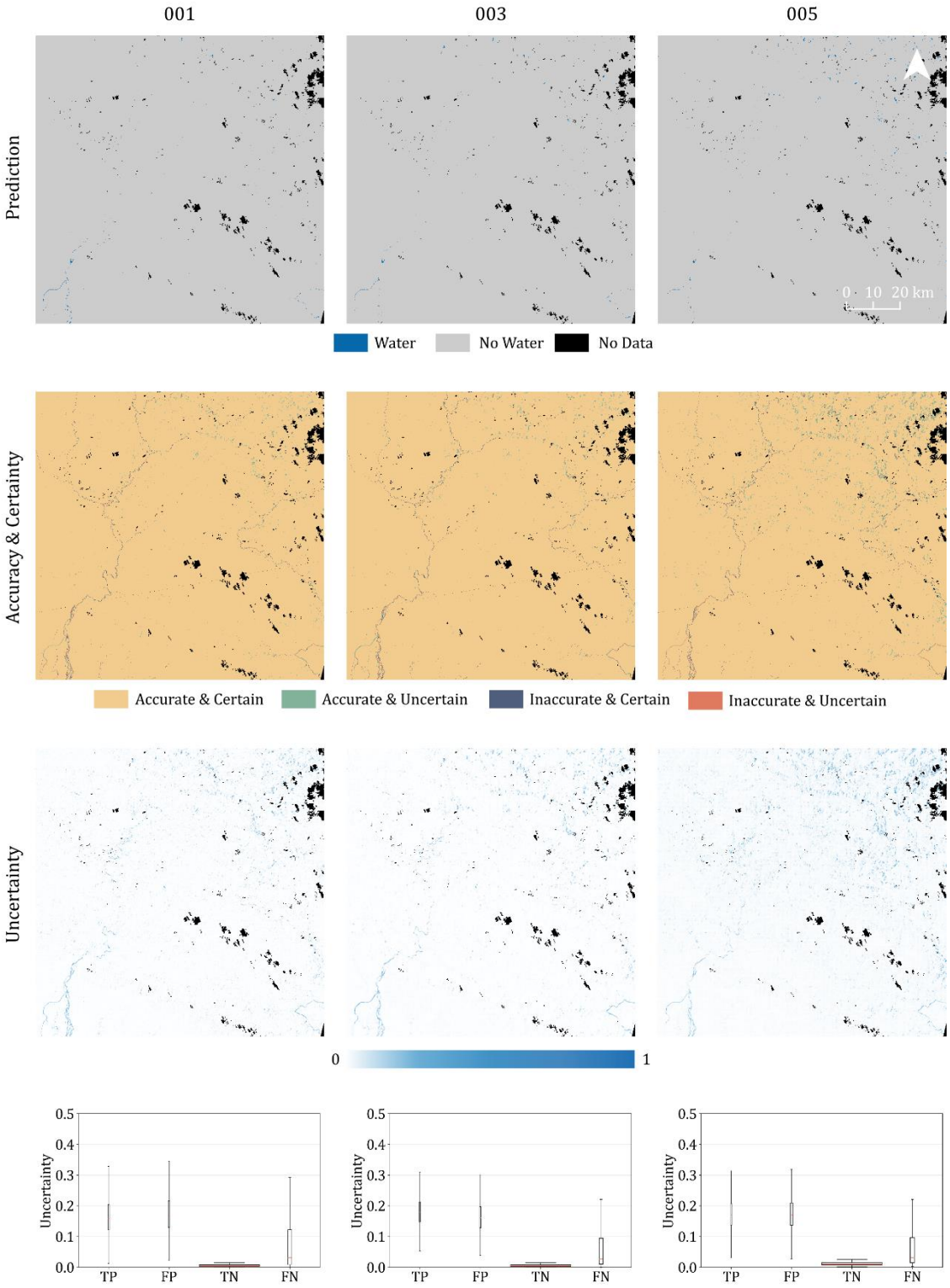
Scene 89, UD1:


001
003
005
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Uncertainty
0
1
Uncertainty
0.5
0.4
0.3
0.2
0.1
0.0
TP
FP
TN
FN

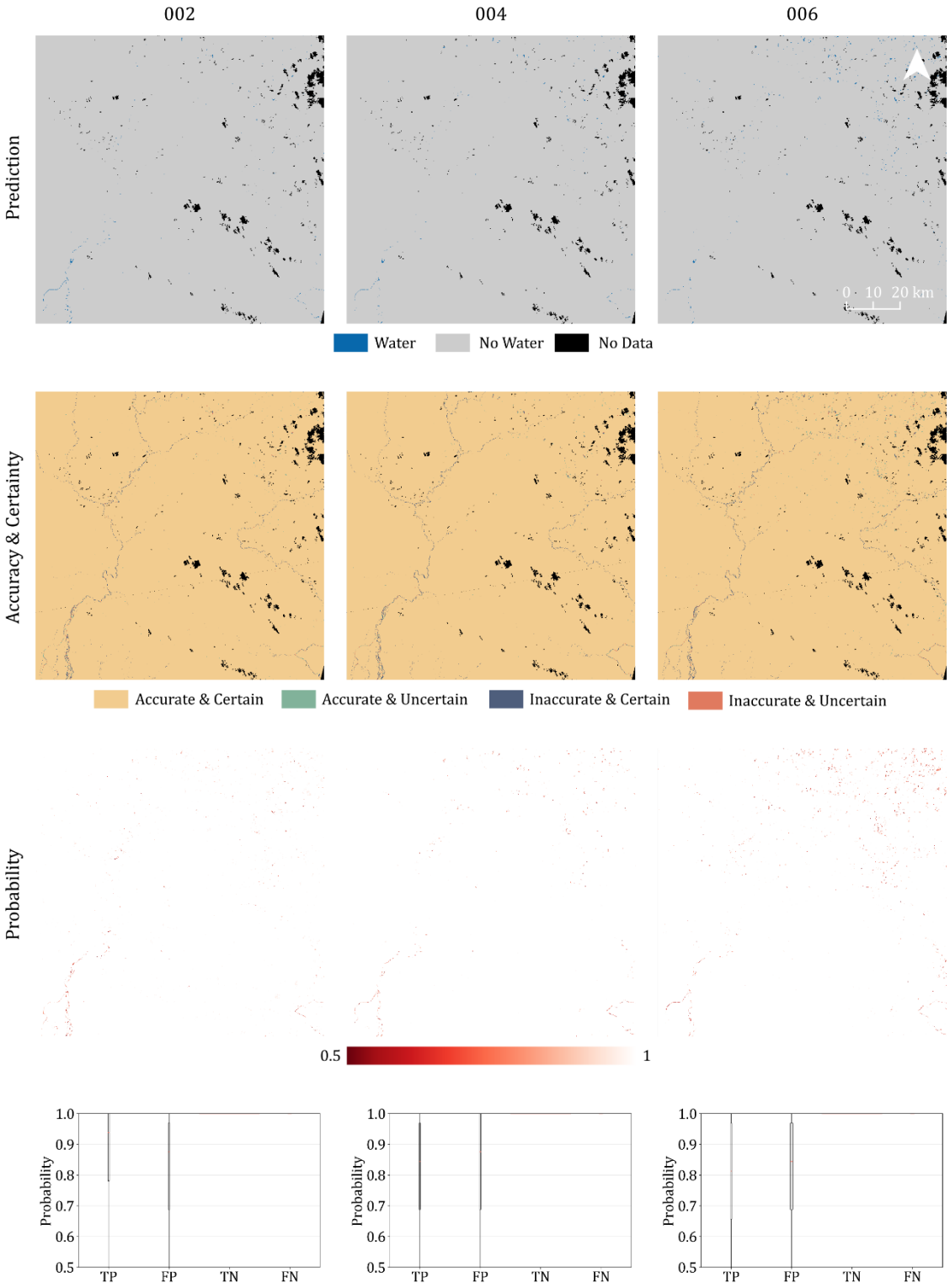
Scene 89, UD2:


002
004
006
Prediction
0 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Probability
0.5
1
Probability
1.0
0.9
0.8
0.7
0.6
0.5
TP
FP
TN
FN

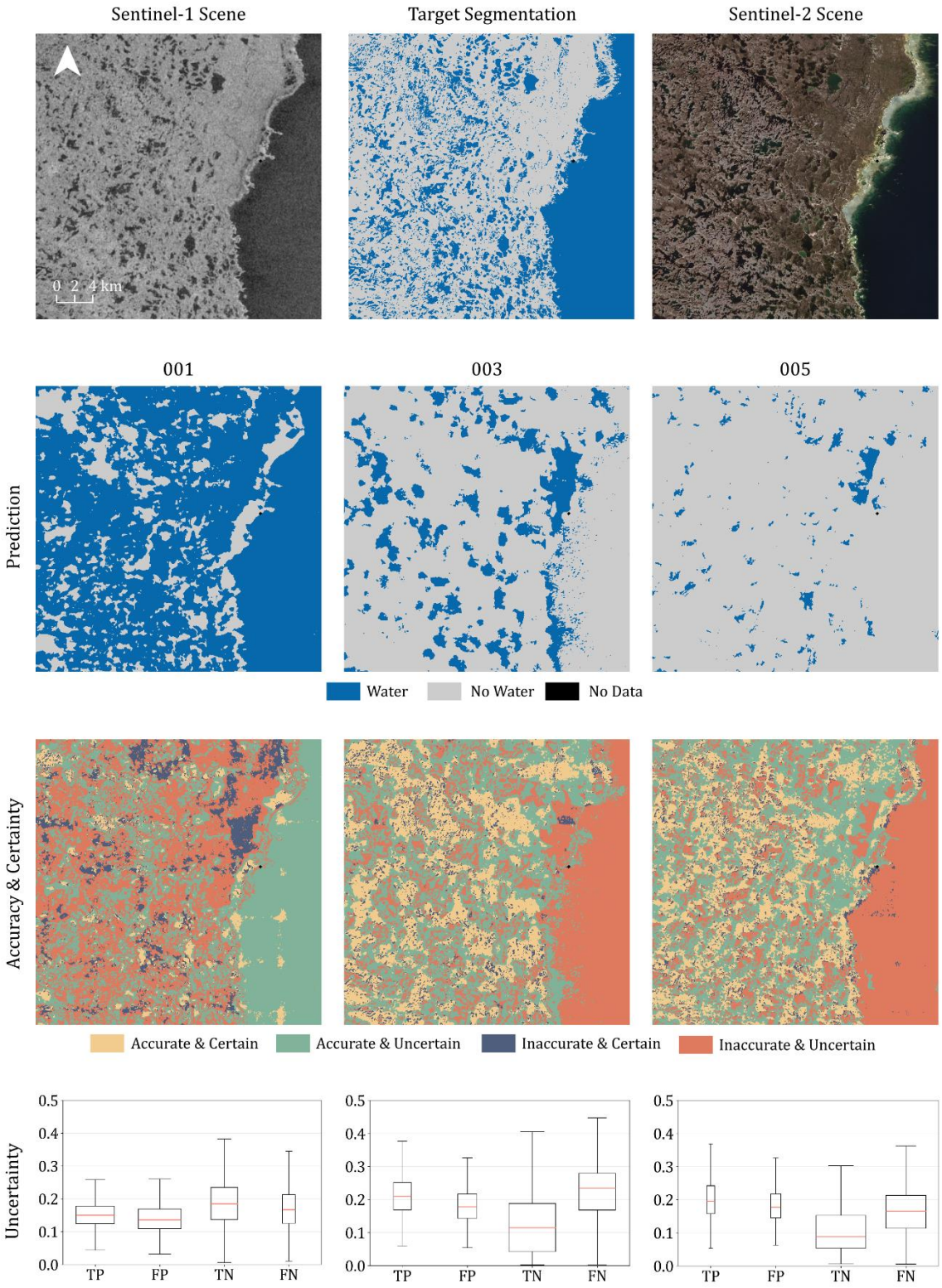
Scene 90, UD1:


001
003
005
Prediction
0 · 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Uncertainty
0
1
Uncertainty
0.5
0.4
0.3
0.2
0.1
0.0
TP
FP
TN
FN

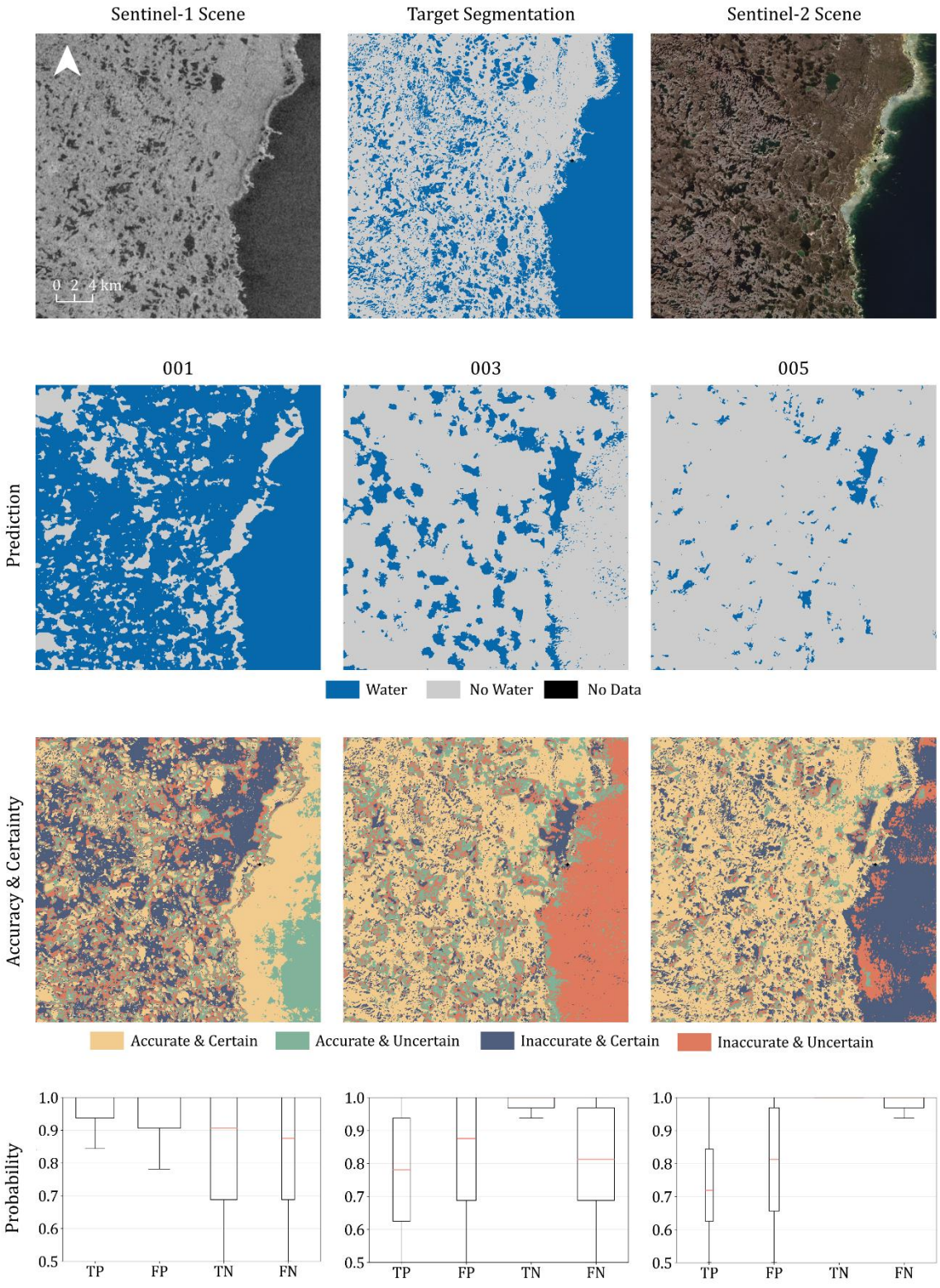
Scene 90, UD2:


002
004
006
Prediction
0 · 10 20 km
Water
No Water
No Data
Accuracy & Certainty
Accurate & Certain
Accurate & Uncertain
Inaccurate & Certain
Inaccurate & Uncertain
Probability
0.5
1
Probability
1.0
0.9
0.8
0.7
0.6
0.5
TP
FP
TN
FN

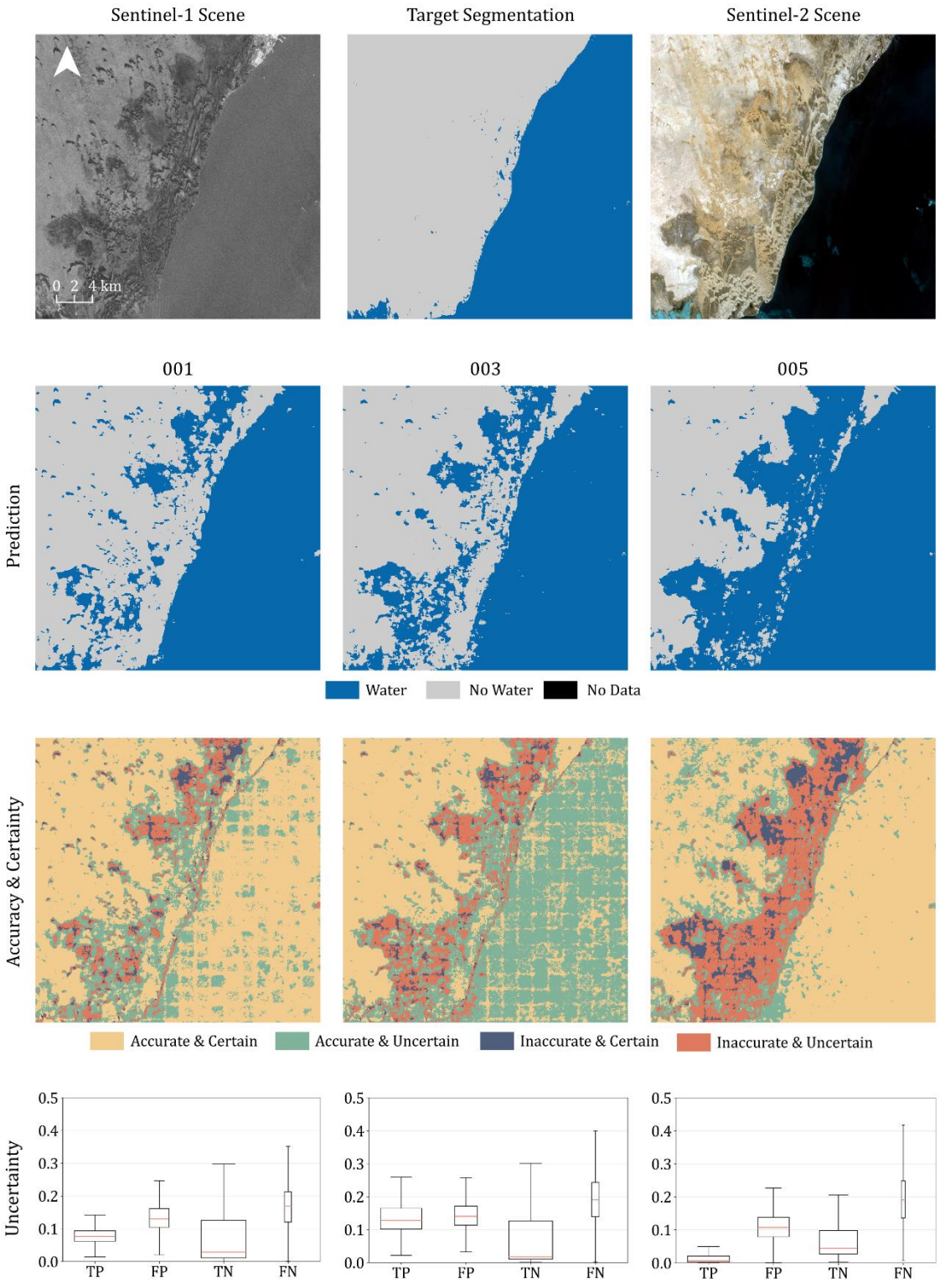
Scene 31, Detail, UD1:



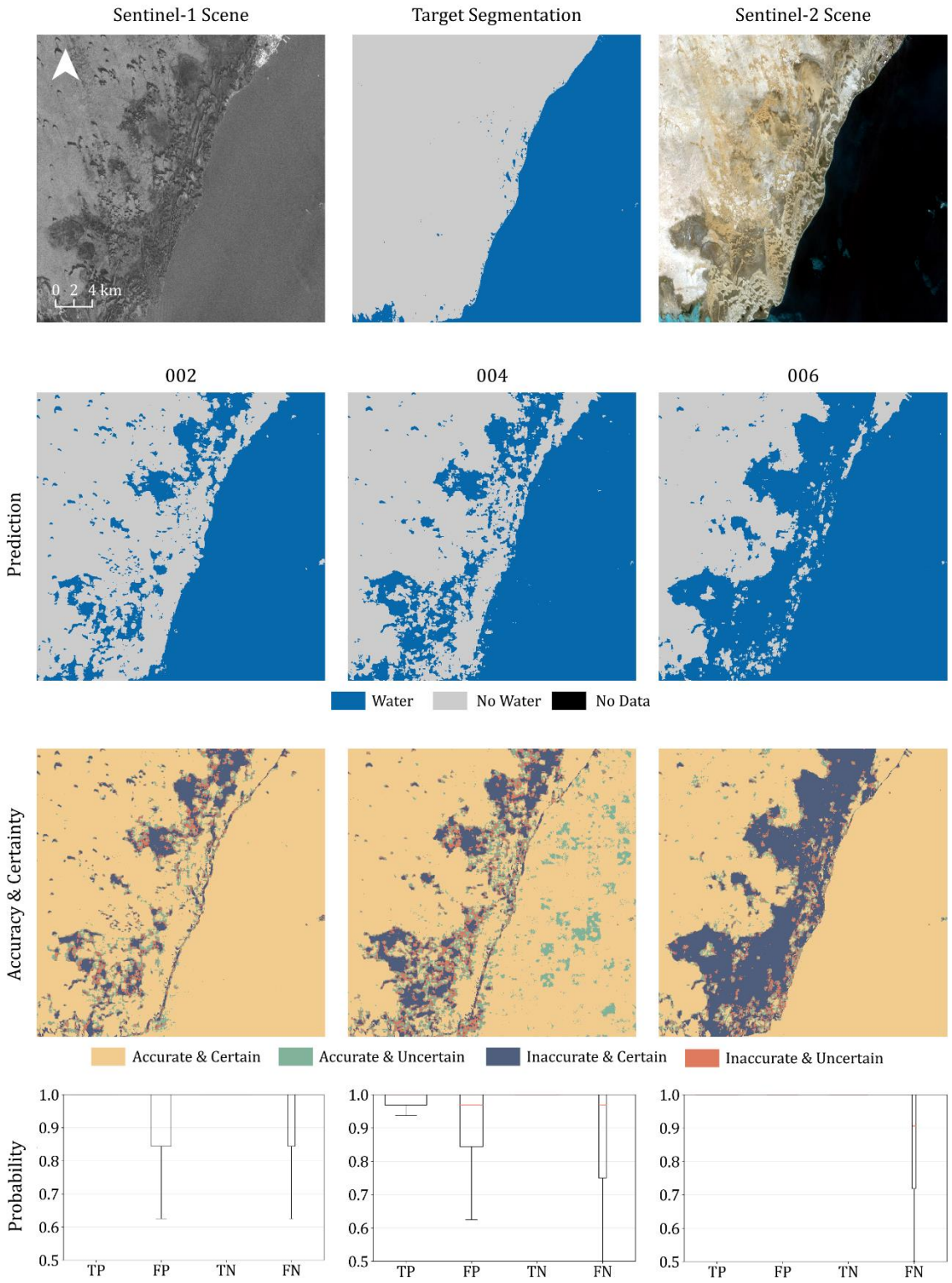
Scene 31, Detail, UD2:



Scene 89, Detail, UD1:



Scene 89, Detail, UD2:



Declaration of Originality

This is to certify, that the master thesis submitted by me is an outcome of my independent and original work. I have duly acknowledged all the sources from which the ideas and extracts have been taken. The project is free from any plagiarism and has not been submitted elsewhere for publication.

Eichstätt, 31.08.2022 P. Medevec

Place, Date, Signature