SELF-TRAINING FOR SEMI-SUPERVISED DEEP CONTOUR DETECTION OF SURFACE WATER

AbdulRahman Alsamman[a], Mohammad Baqiri Syed[a] *

[a]Department of Electrical and Computer Engineering, University of New Orleans – (aalsamma, mbsyed)@uno.edu

KEY WORDS: Semi-supervised, Self-Training, RGB detection, Water Contour Detection, Deep Learning

ABSTRACT:

Contour detection is better for monitoring dynamic and long-term changes to surface water bodies. For that purpose, we present a semi-automated method for collecting and labeling water contours from Landsat-8 and Sentinel-2 images. Due to the need for human inspection, the method has thus far generated 14K labeled images from more than 1.5M images. Given the cost of data labeling, we propose a deep semi-supervised self-learning system performed in two training stages, known as teacher-student. The teacher is trained on the accurate human-labeled data, then used to pseudo label the remaining unlabeled data. The student is trained on both human-labeled and machine pseudo-labeled data. For both teacher and student, we use a uniquely designed multiscale UNet classifier that uses fewer parameters and is more accurate than other state-of-the-art classifiers. Random augmentations are used to “noise” the student model and improve its generalization, and normalization schemes are used to blend the human-labeled loss with the machine-labeled loss. Comparisons to existing water body detection classifiers and segmentation classifiers show the superiority of our proposed system in detecting water contours.

1. INTRODUCTION

Monitoring surface water from remote sensing data is a critical GIS task for risk evaluation, resource management, public policy, emergency response, cartography, and education. Many remote sensing technologies (Huang et al., 2018) are currently available, providing data that vary in cost, temporal resolution, spatial resolution, spectral resolutions, and the number of spectral channels.

Surface water monitoring techniques (Gao, 1996; Xu, 2006; Fisher et al., 2016; Feyisa et al.,2014; Wang et al., 2018; Friedl & Brodley, 1997; Mueller et al., 2016; Aung & Tint, 2018; Cordeiro et al., 2021; Isikdogan et al., 2017; Isikdogan et al., 2020) have focused on the multispectral detection of water bodies that is sensitive to the infra-red (IR) channels. In the planar view, contour detection is more effective in capturing dynamic and long-term changes to surface water than water body detection. Additionally, the dependence on the IR channels makes the detectors expensive and requires recalibration of the system to IR sensing technology (bandwidth, central wavelength, sensitivity, etc.) We propose RGB-based detection that – much like humans – can detect contours without relying on multispectral data.

To aid in this effort, we have started collecting satellite data representing a variation of Landsat and Sentinel waterbody images (lakes, rivers, shores, etc.) from across the globe. We employed rule-based metrics and basic image processing to label the contour data and used visual (human) inspection to isolate and remove inaccurately labeled portions. The process has been extremely slow, thus far yielding only 14K useful images from over 200K candidates, with over 1M images still unchecked.

Given the cost of data labeling, we propose to use a deep semi-supervised self-learning framework in which our unique multiscale UNet-style classifier is trained on a small subset of the labeled data. The trained classifier, also known as a teacher model, is then used to pseudo-label the more extensive set of unlabeled data. Then the classifier is retrained with both human and pseudo-labeled data to achieve a more robust classifier, known as the student model. During the student model training, 50% of each batch is randomly selected from the human-labeled data, and the human-labeled loss is weighted more heavily than the machine-labeled loss. This is done to prevent the pseudo-labeled data from dominating the learning process. The student model batch also undergoes random augmentations of vertical flip, horizontal flip, and rotation to make it “noisier.” The training process reiterates, with the student model becoming the new teacher. We found that after three iterations, the performance improvements become negligible.

In the proceeding sections, we will describe our data collection process (section 2), the architecture of our unique multiscale classifier (section 3), semi-supervised self-training (section 4), and experimental results (in section 5) that demonstrate the superiority of the proposed classifier.

2. DATA COLLECTION AND LABELLING

1.1. Collection

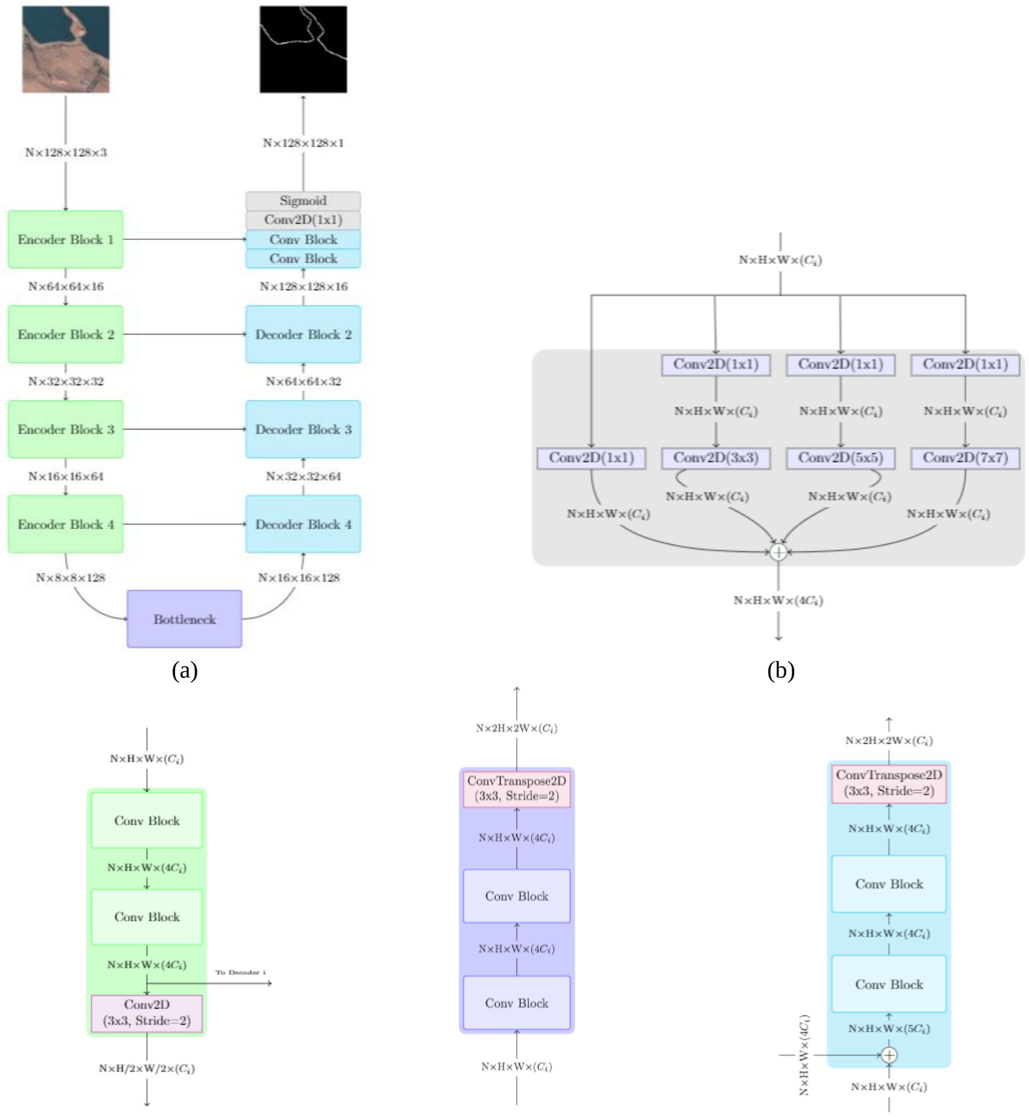
We collected data for both Landsat-8 and Sentinel-2 satellites. There are two datasets that we created for each of the satellite data. One was fully supervised training the second was for semi-supervised (self-training).

For the Landsat-8 data collection, a single method was used. The shapefile from DeepWaterMapV2 (Isikdogan et al., 2020) was used to determine potential global water body locations. Using the metadata in the shapefile, locations with any less than 1% water were removed. Google earth engine (GEE) was used to download the data.

.

* Corresponding Author – (mbsyed@uno.edu)

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLIII-B3-2022-1393-2022 | © Author(s) 2022. CC BY 4.0 License.



(a) (b)

(c) (d) (e)

Figure 1. UNET Architecture, (a) full architecture and (b) encoder, (c) decoder, (d) bottleneck, and (e) conv blocks. The ⊕ operator represents the channel concatenation operation.

Two methods were used for Sentinel-2 data collection, one for supervised learning and one for semi-supervised learning. For supervised learning, a shapefile from BlueDotWater[1] was used to determine the locations of inland water bodies. These were downloaded using Sentinel's python API. For semi-supervised learning, data was collected using the same method used in Landsat-8 data collection. The satellite images were labeled using NDWI (Gao, 1996) to detect water bodies. The water contour is then labeled by subtracting NDWI from its morphological image dilation. The process yield many inaccuracies in the contour even within the same image. To improve the yield, the satellite images were split into 128 × 128 tiles. Human inspection is then used to identify accurately labeled tiles from inaccurate ones.

Currently, we have two sets of data in our repository[2,] one for unlabeled and the second for labeled data. There are over 1M+ Landsat tiles and 1.4M+ unlabeled Sentinel tiles. To make sure that the used data has a contour, we eliminate tiles with with less than 1% water. This means than only 490,070 Landsat tiles and 400,682 Sentinel tiles are used for unlabeled self-training from the unlabeled dataset for Landsat and Sentinel.

The labeled data was hand-selected from the unlabeled data set. This is an extremely slow process with a minimal return. 200,000 images were visually inspected to create a labeled dataset containing 7,000 tiles for Landsat and 7,174 images for Sentinel. We balanced the dataset to avoid an abundance of water-only or land-only tiles.

Each tile in the datasets is stored as 16-bit raw satellite data with six channels in the following order: blue (b1), green (b2), red (b3), NIR (b4), SWIR1 (b5), SWIR2 (b6). As we will be working with RGB data, we convert the raw data into True color images (TCI). For Landsat, we recommend subtracting the min and dividing by the max. For Sentinel, clip it at 3558 first and then divide by the same number. Each image has metadata, including

(1) www.blue-dot-observatory.com/
(2) https://github.com/mbsyed/Deep-Surface-Water-Contour-Detection

satellite source and water percentage for each image. The data also contains JRC (Pekel et al. 2016) water labels for each Landsat tile for reference.

3. Multiscale UNet

Our proposed UNet-based water contour detector can be seen in Figure 1. Our architecture design has encoder/decoder layers that are based on the multiscale convolution block seen in Figure 1(b). Our model uses multiscale 2D filters that are effective in capturing contours. The 1x1 filters are effective in controlling data expansion and help weigh the channels going forward. Each convolution has a Batch Normalization (BN) layer before it to avoid outlier data in a batch.

We chose to use a stirded convolution instead of max-pooling for the down-sampling process. This adds a few parameters to the architecture but the overall performance increases. “Skip” connections between corresponding encoder and decoder blocks are a general attribute of UNet systems that have been shown to improve training and provide better localization in the output. A sigmoid output is used to classify each pixel output as a contour or non-contour (i.e., 1 or 0).

4. Semi-Supervised Self-Training

Supervised vs. unsupervised learning models are determined by the labeled vs. the unlabeled data used for learning. Semi-supervised learning (SSL) aims to combine labeled and unlabeled data to improve the learning task. SSL consists of a variety of techniques (van Engelen & Hoos, 2019) that can be generalized as one of two scenarios: either the system is a supervised learning model that benefits from unlabeled data (inductive) or an unsupervised learning model that is improved by labeled data (transductive).

Self-learning (aka self-training or wrapper methods) is an inductive SSL that aims to train a classifier on a small, accurately (human) labeled set of data. The trained classifier is then used to pseudo-label a larger unlabeled data set. The accurately labeled data and the machine pseudo-labeled data are then combined to train a new classifier. The two classifier stages are sometimes referred to as teacher-student models.

A variety of approaches for self-training have been proposed, see (Triguero et al., 2013) for a review. These vary in the number of classifiers used, the type of classifiers, how the pseudo labeled data is incorporated into retraining, and how many iterations of teacher-student training cycles are used. Implementations such as (Yalniz et al., 2019) use a more powerful teacher model, while others such as (Xie et al., 2020) use a more powerful student model. In (Xie et al., 2020; (Zoph et al., 2020)) noise is added to student model data in the form of random augmentations to help improve the system’s generalization. In (Sohn et al., 2020; Tang et al., 2021), the pseudo-labeled data is ranked prior to student model training.

4.1. Proposed Training Process: Teacher Model

We start with 7,000 Landsat, and 7,174 Sentinel true-color RGB images and their corresponding accurately labeled contours. Each dataset is randomly split into training/testing. Landsat has 5,000 images for training and 2,000 images for testing. Sentinel has a few more images, with 5,121 for training and 2,053 for testing. The teacher model is our multiscale UNet that is trained using an Adam optimizer (learning rate of 0.003, beta1 is set at 0.9, and beta2 is 0.999). The maximum batch size that our GPU can support is 64. We allowed the model to train for 50 epochs. We used a combination of three loss functions, Binary cross-entropy (BCE), Dice, and IoU. BCE captures the pixel-level loss in the image, and intersection over union (IoU) and Dice loss are used to capture contour object-related loss. All three losses are combined equally.

$$L = L_{BCE} + L_{IoU} + L_{Dice} \qquad (1)$$

Due to the significant imbalance between contour and non-contour pixels, the pixels are first weighted by the ratio of non-water pixels to the number of water pixels. Additionally, a $n \times n$ border makes errors closest to the contour count more heavily than those outside the $n \times n$ border. We found a border of 9×9 to be optimum.

4.2. Proposed Training Process: Student Model

The trained teacher model is used to provide machine pseudo-labels for 490,070 unlabeled Landsat and 400,682 unlabeled Sentinel images. Our multiscale UNet is selected again as the student model and retrained from scratch using both human and machine labels. Due to the large number of pseudo-labels to human labels, half the batch (32) is randomly sampled from the human-labeled data, while the other half is sampled from the pseudo-labeled data. Additionally, 50% of the entire batch is randomly selected for augmentation to add noise to the system. When an image is selected for augmentation, one of four augmentations is randomly chosen: vertical flip, horizontal flip, and ±90° rotations. In each training batch, the loss from human and pseudo labeled data is normalized as such

$$\hat{L} = \frac{1}{1+\gamma}\left(L_h + \gamma \frac{\overline{L_h}}{\overline{L_p}} L_p\right) \qquad (2)$$

where L_h and L_p are the human-labeled loss and the pseudo-labeled loss for that batch, while $\overline{L_h}$ and $\overline{L_p}$ are the exponential moving average losses with a decay rate of 0.9997. The weight rate ($\gamma = 3$) was found to be optimum. The student model was trained for nine epochs for Landsat data (7.7K iterations) and eight epochs (6K iterations) for Sentinel.

4.3. Proposed Training Process: Iterations 2 and 3

Following the student model training, we convert the student model into a teacher model and use it to pseudo-label the unlabeled data again. The multiscale UNet is reinitialized, and step 4.2. is repeated. Due to a large number of training batches, we employ a cosine annealing schedule for the learning rate.

5. RESULTS

Tables 1 and 2 compare our multiscale UNet architecture against other popular object segmentation systems found in the literature. DeeplabV3+ (Chen et al. 2018), UNet- Resnet (Ronneberger et al. 2015), UNet++ (Zhou et al. 2018), and PAN (Li et al. 2018) are popular DL-based segmentation techniques. DWM is a DL waterbody detection model that was retrained on our data specifically for contour detection. Waterdetect (Cordeiro et al. 2021) is also a water body detector, but it relies on hierarchical clustering of rule-based metrics.

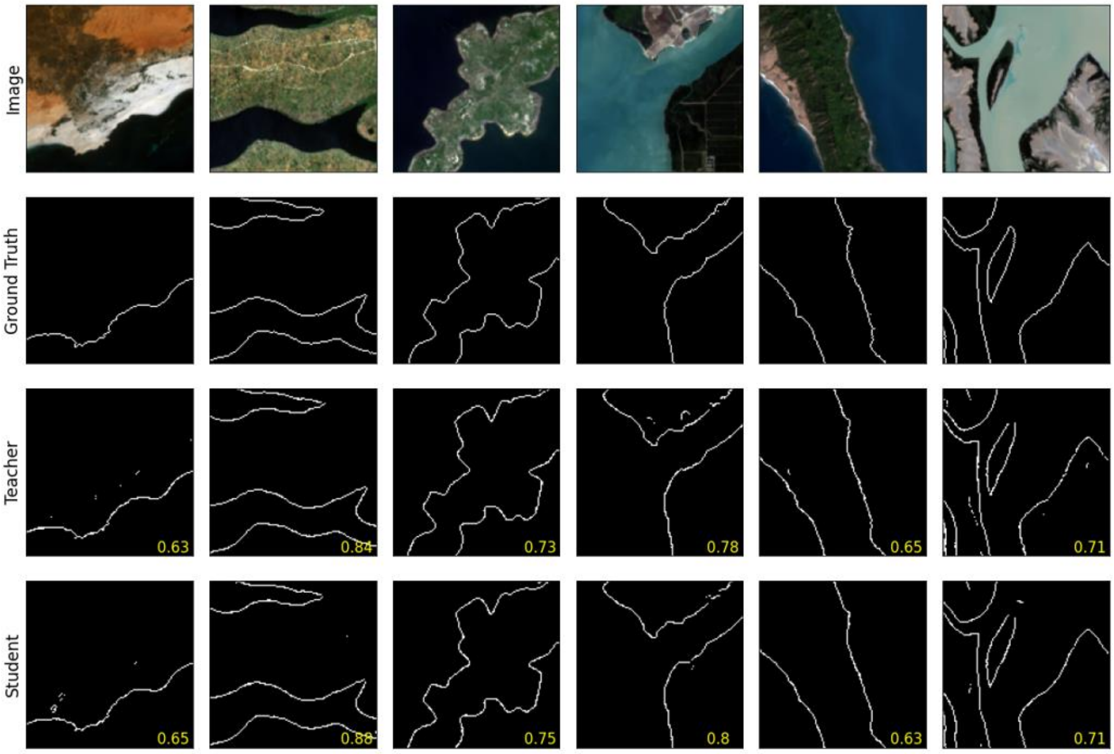


Figure 2. Results on Landsat data. The rows are RGB image, ground truth, teacher model's prediction, and student model's prediction. The F-score for an individual image can be in the bottom right corner.

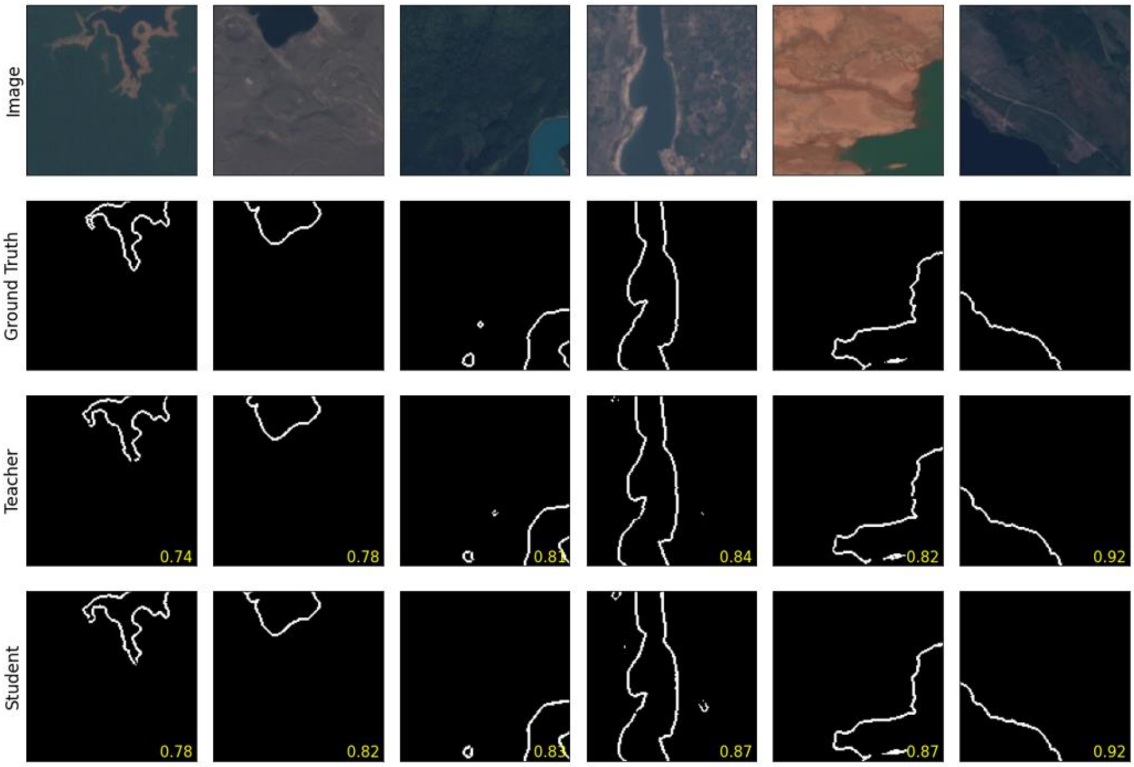


Figure 3. Results on Sentinel data. The rows are RGB image, ground truth, teacher model's prediction, and student model's prediction. The F-score for an individual image can be in the bottom right corner.

Model	Time (per iter.)	Param (M)	Landsat	
			F-score	AP
DeepLabV3+	21.4	26.68	0.4586	0.3386
UNet-Resnet	23.5	32.52	0.6428	0.6435
PAN	20.7	24.26	0.4361	0.3261
UNet++	60.7	48.99	0.6579	0.6559
DWM	17	37.21	0.5583	0.4377
WaterDetect	N/A	N/A	0.6025	NA
UNet-Multiscale (Ours)	44	22.85	0.6920	0.7106

Table 1. Comparison of Water Contour Detection using our RGB dataset for Landsat.

Model	Time (per iter.)	Param (M)	Sentinel	
			F-score	AP
DeepLabV3+	24.7	26.68	0.5739	0.5125
UNet-Resnet	26.2	32.52	0.6683	0.7068
PAN	23.6	24.26	0.5297	0.4677
UNet++	62.9	48.99	0.6634	0.7059
DWM	19	37.21	0.6237	0.5806
WaterDetect	NA	NA	0.6436	NA
UNet-Multiscale (Ours)	47	22.85	0.7379	0.7980

Table 2. Comparison of Water Contour Detection using our RGB dataset for Sentinel.

Model	Time	Iteration	Landsat	
	epoch/s		F-score	AP
UNet-Multiscale (Ours)	43s	NA	0.6920	0.7106
	410m	1	0.7239	0.7587
		2	0.7337	0.7695
		3	0.7358	0.7754

Table 3. Self-training for Landsat-8

Model	Time	Iteration	Sentinel	
	epoch/s		F-score	AP
UNet-Multiscale (Ours)	47s	NA	0.7379	0.7980
	330m	1	0.7538	0.8193

Table 4. Self-training for Sentinel-2

All systems are trained with our RGB data for accurate water contour detection. The results indicate that our base system uses fewer parameters, has a faster training time, and is more accurate at detecting water contours.

Tables 3 and 4 contain the results of self-training for Landsat and Sentinel, respectively. A clear improvement can be seen in the model's performance, where there is a 2% improvement for Sentinel's F-score and a 6% improvement for Landsat. We can also see a clear improvement in the model's output before and after self-training as the F-score for individual images increases.

Although both models were trained for three iterations of teacher-student iterative training, there was no improvement in the performance of the F-score for Sentinel data after the first iteration.

6. CONCLUSION

It is laborious and time-consuming to hand-select training data; we present a self-training technique that enhances our baseline's performance and removes the need to hand-select data. Due to the lack of well-labeled data, we present a dataset that can be used in training deep learning models. We also present a deep learning model that can accurately detect contours faster and use fewer parameters than state-of-the-art segmentation and object detection models.

ACKNOWLEDGMENTS

This research was made possible through funding from the Navy Research Laboratory.

REFERENCES

Huang, C., Chen, Y., Zhang, S., & Wu, J. (2018). Detecting, extracting, and monitoring surface water from space using optical sensors: A Review. Reviews of Geophysics, 56(2), 333–360. doi.org/10.1029/2018rg000598

Gao, B.-cai. (1996). Ndwi—a normalized difference water index for remote sensing of vegetation liquid water from space. Remote Sensing of Environment, 58(3), 257–266. doi.org/10.1016/s0034-4257(96)00067-3

Xu, H. (2006). Modification of normalised difference water index (NDWI) to enhance open water features in remotely sensed imagery. International Journal of Remote Sensing, 27(14), 3025–3033. doi.org/10.1080/01431160600589179

Fisher, A., Flood, N., & Danaher, T. (2016). Comparing landsat water index methods for Automated Water Classification in Eastern Australia. Remote Sensing of Environment, 175, 167–182. doi.org/10.1016/j.rse.2015.12.055

Feyisa, G. L., Meilby, H., Fensholt, R., & Proud, S. R. (2014). Automated Water Extraction Index: A new technique for surface water mapping using landsat imagery. Remote Sensing of Environment, 140, 23–35. doi.org/10.1016/j.rse.2013.08.029

Wang, X., Xie, S., Zhang, X., Chen, C., Guo, H., Du, J., & Duan, Z. (2018). A Robust Multi-band water index (MBWI) for automated extraction of surface water from Landsat 8 Oli imagery. International Journal of Applied Earth Observation and Geoinformation, 68, 73–91. doi.org/10.1016/j.jag.2018.01.018

Friedl, M. A., & Brodley, C. E. (1997). Decision Tree Classification of land cover from remotely sensed data. Remote Sensing of Environment, 61(3), 399–409. doi.org/10.1016/s0034-4257(97)00049-7

Mueller, N., Lewis, A., Roberts, D., Ring, S., Melrose, R., Sixsmith, J., Lymburner, L., McIntyre, A., Tan, P., Curnow, S., & Ip, A. (2016). Water observations from space: Mapping Surface Water from 25 years of landsat imagery across Australia. Remote Sensing of Environment, 174, 341–352. doi.org/10.1016/j.rse.2015.11.003

Moh Aung, E. M., & Tint, T. (2018). Ayeyarwady River regions detection and extraction system from Google Earth imagery. 2018 IEEE International Conference on Information Communication and Signal Processing (ICICSP). doi.org/10.1109/icicsp.2018.8549806

Cordeiro, M. C. R., Martinez, J.-M., & Peña-Luque, S. (2021). Automatic water detection from multidimensional hierarchical clustering for sentinel-2 images and a comparison with level 2A processors. Remote Sensing of Environment, 253, 112209. doi.org/10.1016/j.rse.2020.112209

Isikdogan, F., Bovik, A. C., & Passalacqua, P. (2017). Surface water mapping by Deep Learning. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 10(11), 4909–4918. doi.org/10.1109/jstars.2017.2735443

Isikdogan, L. F., Bovik, A., & Passalacqua, P. (2020). Seeing through the clouds with deepwatermap. IEEE Geoscience and Remote Sensing Letters, 17(10), 1662–1666. doi.org/10.1109/lgrs.2019.2953261

Carroll, M. L., Townshend, J. R., DiMiceli, C. M., Noojipady, P., & Sohlberg, R. A. (2009). A new global raster water mask at 250 m resolution. International Journal of Digital Earth, 2(4), 291–308. doi.org/10.1080/17538940902951401

Homer, C., Huang, C., Yang, L., Wylie, B., & Coan, M. (2004). Development of a 2001 national land-cover database for the United States. Photogrammetric Engineering & Remote Sensing, 70(7), 829–840. doi.org/10.14358/pers.70.7.829

Verpoorter, C., Kutser, T., Seekell, D. A., & Tranvik, L. J. (2014). A global inventory of lakes based on high-resolution satellite imagery. Geophysical Research Letters, 41(18), 6396–6402. doi.org/10.1002/2014gl060641

Yamazaki, D., Trigg, M. A., & Ikeshima, D. (2015). Development of a global ~90m water body map using multi-temporal landsat images. Remote Sensing of Environment, 171, 337–351. doi.org/10.1016/j.rse.2015.10.014

Prigent, C., Papa, F., Aires, F., Jimenez, C., Rossow, W. B., & Matthews, E. (2012). Changes in land surface water dynamics since the 1990s and relation to population pressure. Geophysical Research Letters, 39(8). doi.org/10.1029/2012gl051276

Lehner, B., &amp; Döll, P. (2004). Development and validation of a global database of lakes, reservoirs, and wetlands. Journal of Hydrology, 296(1-4), 1–22. doi.org/10.1016/j.jhydrol.2004.03.028

van Engelen, J. E., & Hoos, H. H. (2019). A survey on semi-supervised learning. Machine Learning, 109(2), 373–440. https://doi.org/10.1007/s10994-019-05855-6

Triguero, I., García, S., &amp; Herrera, F. (2013). Self-labeled techniques for semi-supervised learning: Taxonomy, software, and empirical study. Knowledge and Information Systems, 42(2), 245–284. doi.org/10.1007/s10115-013-0706-y

I Zeki Yalniz, Hervé Jégou, Kan Chen, Manohar Paluri, and Dhruv Mahajan. Billion-scale semi-supervised learning for image classification. arXiv preprint arXiv:1905.00546, 2019.

Xie, Q., Luong, M.-T., Hovy, E., &amp; Le, Q. V. (2020). Self-training with noisy student improves ImageNet Classification. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). doi.org/10.1109/cvpr42600.2020.01070

Zoph, Barret & Ghiasi, Golnaz & Lin, Tsung-Yi & Cui, Yin & Liu, Hanxiao & Cubuk, Ekin & Le, Quoc. Rethinking pre-training and self-training. In Advances in Neural Information Processing Systems, 2020.

Huang, G., Sun, Y., Liu, Z., Sedra, D., &amp; Weinberger, K. Q. (2016). Deep Networks with stochastic depth. Computer Vision – ECCV 2016, 646–661. doi.org/10.1007/978-3-319-46493-0_39

Najafi, Amir, Maeda, Shin-ichi., Koyama, Masanori., & Miyato, Takeru., Robustness to adversarial perturbations in learning from incomplete data. In Advances in Neural Information Processing Systems, 2019.

Sohn, K., Berthelot, D., Li, C., Zhang, Z., Carlini, N., Cubuk, E.D., Kurakin, A., Zhang, H., & Raffel, C. (2020). FixMatch: Simplifying Semi-Supervised Learning with Consistency and Confidence. ArXiv, abs/2001.07685. doi.org/10.48550/arXiv.2001.07685

Tang, Y., Chen, W., Luo, Y., & Zhang, Y. (2021). Humble teachers teach better students for semi-supervised object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 3132-3141).

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLIII-B3-2022-1393-2022 |