COMPARISON OF K-NEAREST NEIGHBOR, MULTIPLE LINEAR REGRESSION, AND RANDOM FOREST CLASSIFIERS FOR DEPTH EXTRACTION IN THE SHALLOW WATER OF KEPULAUAN SERIBU, INDONESIA

(Perbandingan Klasifikasi K-Nearest Neighbor, Multiple Linear Regression, dan Random Forest untuk Ekstraksi Data Kedalaman di Perairan Dangkal Kepulauan Seribu, Indonesia)

Rifqi Muhammad Harrys, Ratna Sari Dewi, Teguh Sulistian, Dimas Hanityawan Suryopuspito, Fajar Triady Mugiarto
Geospatial Information Agency of Indonesia
Jl. Raya Jakarta-Bogor KM 46 Cibinong 16911 Jawa Barat, Indonesia
E-mail: rifqi.muhammad@big.go.id

Diterima: 03 Februari 2022; Direvisi: 22 Mei 2023; Disetujui untuk Dipublikasikan: 29 Mei 2023

ABSTRACT

Satellite-derived bathymetry is a method used to overcome the limitation of survey vessels when acquiring depth data in shallow waters of less than 2 m, especially depths of 0-2 m. Currently, the SDB method has been widely used to provide shallow water bathymetric data. Besides this method can provide wide coverage of depth data, the availability of multitemporal and multiresolution images allows the method to be categorized as a relatively low-cost method compared to conventional surveys. This study compares SDB methods in deriving depth data using various machine learning algorithms using Sentinel-2A images in Kepulauan Seribu, Indonesia. Three machine learning algorithms were compared, namely K-Nearest Neighbors (KNN), Multiple Linear Regression (MLR), and Random Forest (RF), to observe the best-performing method. SDB was applied by combining echo-sounding measurements and the reflectance of blue, green, red, and near-infrared bands of Sentinel 2A. Our research revealed that RF provided the best accuracy compared to MLR and KNN. However, the resulted depth range could not cover very shallow water depth at 0 m. Only the MLR could detect zero depth, but it has the worst RMSE value. KNN provided a feasible result with slightly higher RMSE compared to RF, nonetheless, it took longer runtime for about 30% higher than RF.

Keywords: bathymetry, k-nearest neighbors, machine learning, multiple linear regression, random forest, satellite derived bathymetry

ABSTRAK

Satellite derived bathymetry adalah suatu metode yang dapat digunakan untuk mengatasi kelemahan kapal survey ketika mengakuisisi data kedalaman pada perairan dangkal yang kurang dari 2 m, khusus nya kedalaman 0-2 m. Saat ini, metode SDB sudah banyak dimanfaatkan untuk penyediaan data batimetri perairan dangkal. Selain metode ini mampu menghasilkan cakupan data kedalaman yang luas, ketersediaan citra multitemporal dan multiresolusi memungkinkan metode ini dikategorikan sebagai metode yang relatif murah dibandingkan dengan survey konvensional. Studi ini fokus membandingkan metode SDB dalam menurunkan data kedalaman dengan menggunakan berbagai algoritma pemelajaran mesin dengan menggunakan citra Sentinel 2A di Kepulauan Seribu, Indonesia. Tiga algoritma pemelajaran mesin dibandingkan yaitu K-Nearest Neighbors (KNN), Multiple Linear Regression (MLR), dan Random Forest (RF) untuk mengamati metode dengan performa terbaik. SDB diterapkan dengan menggabungkan hasil pengukuran echo-sounding *dan nlai reflektan band biru, hijau, merah, dan inframerah dekat citra Sentinel 2A. Penelitian ini mengungkapkan bahwa RF memberikan akurasi terbaik dibandingkan dengan MLR dan KNN, namun kisaran kedalaman yang dihasilkan tidak dapat mencakup kedalaman air yang sangat dangkal 0 m. Hanya MLR yang dapat mendeteksi kedalaman 0 m, tetapi algoritma tersebut memiliki nilai RMSE terburuk. Faktanya, KNN memberikan hasil yang cukup layak dengan RMSE yang sedikit lebih tinggi dibandingkan dengan RF, namun membutuhkan waktu proses yang lebih lama sekitar 30% dari pada RF.*

***Kata kunci**: batimetri, k-nearest neighbors, pembelajaran mesin, multiple linear regression, random forest satellite derived bathymetry*

INTRODUCTION

Bathymetric surveying by conventional survey vessels is exceedingly complex and dangerous in shallow waters (Caballero & Stumpf, 2020). Moreover, this technique is time-consuming, demanding, and expensive (Duplančić Leder, Leder, & Peroš, 2019). However, bathymetric data in shallow waters is required to create a precise coastal Digital Elevation Model (DEM).

An alternative method to extract bathymetric data in shallow waters is using Satellite Derived Bathymetry (SDB). The SDB method uses satellite imagery for depth data extraction (Marks, 2018). However, depth extraction by using SDB generally works well in very shallow water areas with clear waters (Caballero & Stumpf, 2020; Dewi et al., 2019). This study explored the SDB algorithm, which can produce high-precision bathymetric data.

There are two approaches in the SDB, namely analytical and empirical approaches. Empirical approach is based on the relationship between image pixel values and in-situ depth data (Hamylton, Hedley, & Beaman, 2015; Rossi, Mammi, & Pelliccia, 2020). While the analytical approach is based on the principle of light propagation in water that requires a number of optical properties of water as input (Gao, 2009).

As an empirical approach, the satellite-derived bathymetry requires in-situ data as training data, thus water depth measurement is needed. On the other hand, a bathymetric survey using a regular survey vessel has limitations in acquiring data in shallow water due to safety reasons (Caballero & Stumpf, 2020). Fortunately, unmanned survey vessels equipped with Single Beam Echosounder (SBES) or Multi Beam Echosounder (MBES) are increasingly available as technology advances. This technological advancement is known as Unmanned Surface Vehicle (USV), as in **Figure 1**.

Figure 1. Unmanned surface vehicle is equipped with single beam echosounder.

Based on our experience in the field, several advantages of using USV exist. First, the vessel has lightweight and small size, so this can prevent the vessel from running aground. However, if it does run aground, it can always be lifted. Second, it is remotely operable; it can acquire data in hardly accessible waters. In this case, by using USV, water depth measurement in shallow water areas is expected to be much easier. Therefore, using an empirical approach can support deriving water depth in shallow water areas.

Machine Learning (ML) is considered a useful approach to developing a general water depth estimation (Sagawa et al., 2019). Furthermore, Sagawa et al. (2019) mentioned that ML has a broad use for predicting any data if one has data as ground truth and its features to train this ML model, including predicting or deriving water depth. Deriving water depth needs water depth samples as ground truth and features correspond to the water depth value. In this case, the selected ML method will fit the ground truth data to the features (Sagawa et al., 2019). By using the ML, a rapid survey with only a few transects can be conducted, thus, this can reduce the budget and time needed for field survey (Tonion et al., 2020).

This research focused on comparing three machine learning algorithms: KNN, MLR, and RF. The methods were tested using Sentinel 2A imagery in Kepulauan Seribu, Indonesia. Comparison between three SDB techniques as well as the accuracy assessment using bathymetric data, were conducted in order to reveal approaches that have the best performance. Thus, the objectives of this study are summarized as follows: to determine the best SDB model; and to compare the performance of KNN, MLR, and RF for their SDB predictions accuracies and runtime.

METHODS

Applying the SDB in this research required in-situ data (i.e., depth sample). The depth sample was acquired in Kepulauan Seribu, Indonesia (**Figure 2**) using USV equipped with SBES. **Figure 2** shows the extent of the research area of interest (AOI). However, the research scope only covers the shallow water areas inside the AOI that are hardly accessible by a conventional survey vessel. This research workflow is shown in **Figure 3**.

The depth data acquisition was conducted over ten days, from June 27th until July 06th, 2021. The data was acquired in shallow water areas inside the AOI (see **Figure 4**). The depth data was then corrected using daily local tide data and tied to a vertical reference (Mean Sea Level). To ensure the quality of the USV surveyed depth data, we calculated its maximum allowable Total Vertical Uncertainty (TVU) using an equation provided by the International Hydrographic Organization (IHO) following their Standards for Hydrographic Surveys, S-44 edition 6.0.0. The TVU is calculated as follows:

$$\boldsymbol{TVU_{max}} = \sqrt{\boldsymbol{a^2} + (\boldsymbol{b} \times \boldsymbol{d})^2} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(1)$$

where:

TVU_{max} = maximum allowable TVU
a = represents that portion of the uncertainty that does not vary with the depth
b = a coefficient representing a portion of the uncertainty that varies with the depth
d = depth

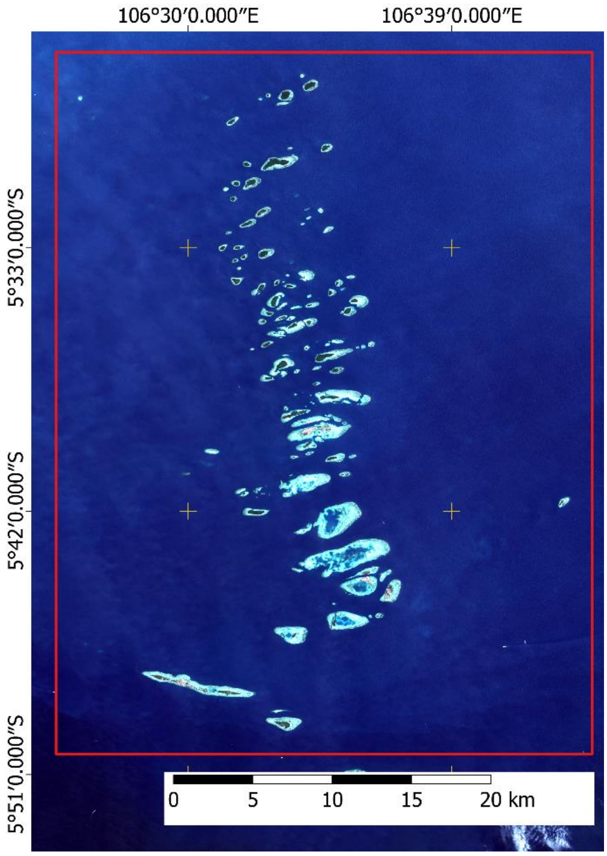


Figure 2. Research area of interest in Kepulauan Seribu, Indonesia shown by Sentinel 2A imagery as a background.

Following Order 2 and Order 1b requirements of the IHO Standard for Hydrographic Surveys, the TVU for Order 2 has a 100% confidence level, and the TVU for Order 1b has 95.12% confidence level. A confidence level over 95% means the depth data quality passes the standards for the safety of navigation hydrographic surveys for Order 2 and Order 1b. Order 2 is areas where a general description of the sea floor is considered adequate. In contrast, Order 1b is areas where under keel clearance is not considered to be an issue for the type of surface shipping expected to transit the area (International Hydrographic Organization, 2020).

Another essential data for Satellite Derived Bathymetry is satellite imagery. The selected satellite imagery for this study was Sentinel 2A because it is a multi-spectral imagery that is free to use and has up to 10-m spatial resolution and a 100 km x 100 km wide exposure scene (Delwart, 2015). Fortunately, one scene of Sentinel 2A can cover all the areas of interest.

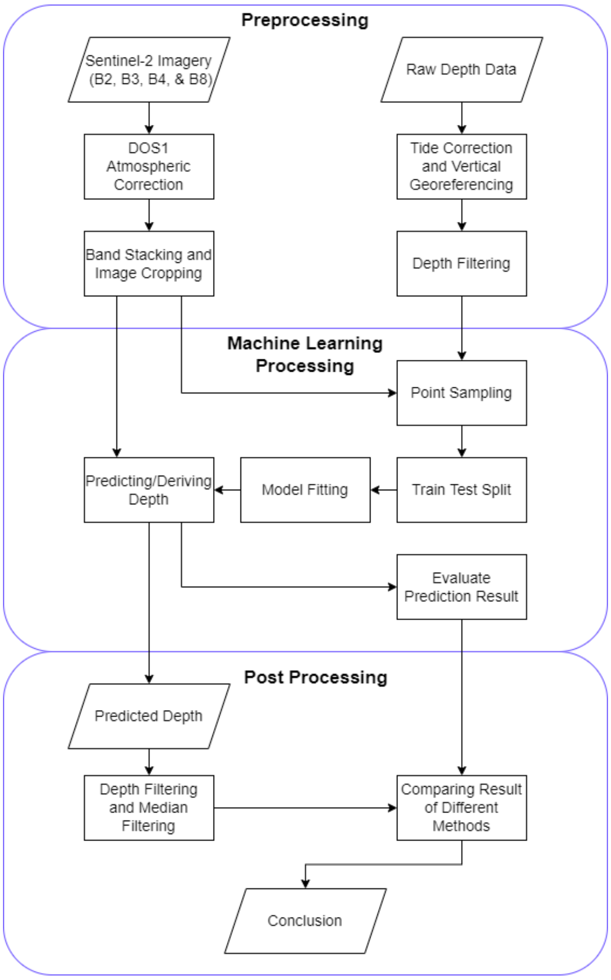


Figure 3. General framework of the research.

A particular Sentinel 2A acquired on August 12th, 2021, was selected because it was cloud-free, especially in shallow water areas, and it has the closest acquisition time with the depth sample data acquisition. The selected imagery then goes through an atmospheric correction, band stacking, and cropping. The Dark Object Subtraction 1 (DOS1) was applied for the atmospheric correction. The DOS1 correction was executed using the Semi-Automatic Classification Plugin -a Python plugin- for the QGIS software developed to facilitate people whose main field is not strictly remote sensing but could benefit from remote sensing analysis (Congedo, 2021).

Sentinel 2A has thirteen bands, however, only four bands (i.e., B2, B3, B4, and B8) were used for this research because they have a 10-meter spatial resolution. Those four bands were stacked into one multiple-band image file and cropped based on the research area of interest. Now that we had the depth sample data as ground truth and four bands of Sentinel 2A imagery as features, we proceeded to derive water depth using three machine learning algorithms. In this research, we used the Scikit-Learn library because it uses high-level language that is easy to use and contains many ML algorithms (Pedregosa et al., 2011).

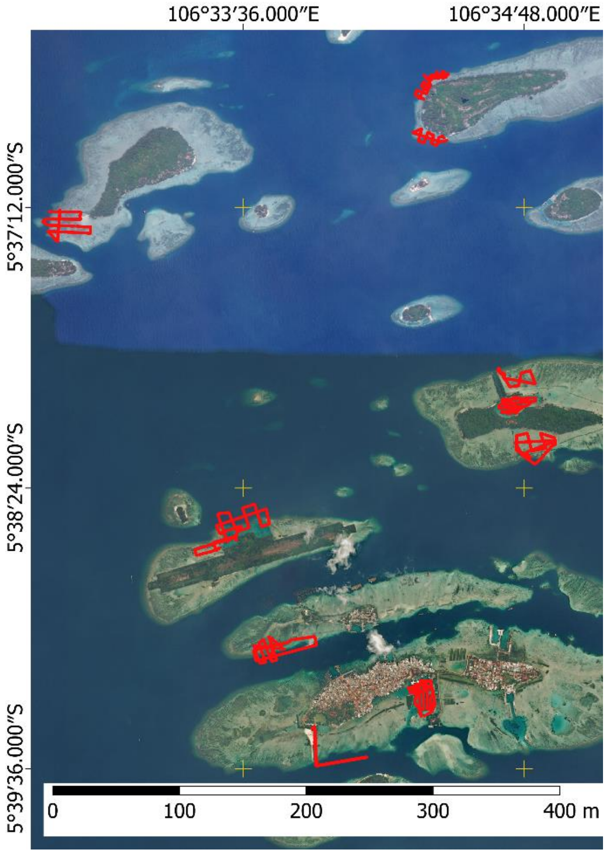


Figure 4. A closer look on some of the depth samples in shallow water areas (using Bing Imagery as the background)

To predict specific outcomes using machine learning from the available data, we need to differentiate between features and labels of the data. In the case of deriving depth using satellite imagery, the label is the depth data itself, and the feature is its attribute represented by raster values of the imagery. With ML, we analyzed the correlation between features and labels.

At this point, we had depth samples and preprocessed imagery. Afterward, raster values of the imagery were extracted and linked to depth values according to its coordinate location, known as point sampling. Before using any ML methods, depth samples and their corresponding raster values were filtered, so we only used depth sample data of 15 m or less. This was because in this research, the target area was limited to very shallow water, and the limitation of this sample data was to reduce the data processing load. In order to apply any of these ML methods, we split the filtered depth samples into training and testing datasets. The training dataset was used for training an ML method, and the testing dataset was used for measuring the accuracy of the prediction of the results. The depth samples and their corresponding raster values were split into training and testing datasets by 75% (train data) to 25% (test data) of all samples randomly.

Three ML methods were compared in this study, namely K-Nearest Neighbors (KNN), Multiple Linear Regression (MLR), and Random Forest (RF). The motivation of the method comparison was to reveal which method performs better. Previous studies mentioned that RF has the best performance (Sagawa et al., 2019) while KNN for SDB was rarely used. So, it is important to find out the performance of KNN against RF and MLR. The KNN method is based on labels of the K-nearest patterns in dataspace (Kramer, 2013). It derives depth values by comparing the training dataset and its corresponding features, finding K-numbers of its closest neighbors, and then predicting the depth from here. MLR extends simple linear regression to include multiple explanatory variables (Tranmer, Murphy, Elliot, & Pampaka, 2020). MLR method derives depth values by solving linear regression using multiple features and the defined training samples. RF is a combination of tree predictors such that each tree depends on the values of a random vector sampled independently with the same distribution for all trees in the forest (Breiman, 2001)

We set certain parameters for each method beforehand and left the rest using a default value from the Scikit-Learn library. In KNN, the parameter is weight. In MLR, we used default values for its parameter because there is not much to adjust. Moreover, in RF, we only set the number of trees parameter. For this research, the number of trees equal to 300 was selected since we obtained a good result after experimenting with several tree parameters in the pre-analysis stage. All the selected parameters are shown in **Table 1**.

Table 1. Selected parameter values

ML Method	Parameter Values
K-Nearest Neighbors	weight = distance
Multiple Linear Regression	All default
Random Forest	Number of trees = 300

Using the training dataset, the ML method was fitted according to its adjusted parameters for each method. When the model fitting is completed, predicting depth using the model becomes possible. The fitted models were used in two processes, deriving depth in every pixel grid of the processed Sentinel 2A imagery (i.e., cropped, stacked, then cropped) and deriving depth in the location of the split test dataset. The former was to derive the depth of the shallow water areas, while the latter was to assess the accuracy of the derived depth by calculating the root mean squared error (RMSE) based on different values of the real depth and the predicted depth of the test dataset..

As the depth samples were filtered, only 15-m of the samples would go through the next processing step. The derived depth as a result of ML processes was also filtered. In other words, the 0–15-meter filter was applied before and after the machine learning process. The logic behind this was that if the input data contained only 0–15-m depth, then the output would also contain only 0–15-m data.

The predicted depth then went through the median filtering process. The median filter is a nonlinear signal processing to replace the noisy value of an image with the median value of its neighbors (Zhu & Huang, 2012). The median filter is useful for removing salt and pepper noise on an image. An image filter usually has a square shape with the size using pixel units and odd numbers (e.g., 3x3, 5x5, 7x7). This filter size means a boundary for which of its neighbors would be included in the median calculation. As the median filter size increases, the filtered image becomes smoother or less sharp. For this reason, using a 3x3 size median filter on the derived depth should be sufficient since using a larger filter would eliminate information detail of the images.

All the ML processes in this study were conducted using Scikit-Learn and other Python-based libraries and all their dependencies, such as NumPy, Pandas, SciPy, Rasterio, and Geopandas. NumPy is useful for manipulating data in an array form (Harris et al., 2020). Pandas is a Python library to process tabular data (McKinney, 2010). SciPy provides fundamental building blocks for solving scientific problems (Virtanen et al., 2020) and in this case is the median filter. Rasterio and Geopandas are tools to process raster and vector data containing geographical information (Gillies, 2013; Jordahl et al., 2020). To make the machine learning processes easier to execute repeatedly, we used a Python-based program based on Satellite Derived Bathymetry (SDB) GUI (Harrys, 2022).

RESULTS AND DISCUSSIONS

There were over 66,032 depth sample points from the bathymetric survey using USV. By taking only 0–15-m of depth data, the used samples were 60,188 points or about 91.15% of all depth samples. The number of depth samples of 3 m is 45,699 or 75.927% of filtered depth samples (see **Table 2**).

Table 2. Filtered depth quantity by depth range

Depth Range (m)	Quantity	
	Amount	Percentage
0 ≤ depth ≤ 3	45,699	75.9
3 < depth ≤ 6	7,513	12.5
6 < depth ≤ 9	2,545	4.2
9 < depth ≤ 12	2,156	3.5
12 < depth ≤ 15	2,275	3.8

Table 3. RMSE and runtime of three machine learning methods

ML Method	RMSE (m)	Runtime (H:M:S)
KNN	0.47	0:01:01.324
MLR	1.79	0:00:04.202
RF	0.46	0:00:39.465

Table 3 shows RMSE in meters and runtime in hours, minutes, and seconds of each ML method mentioned in the previous section. According to **Table 3**, the RF method has the lowest RMSE, whereas MLR has the highest RMSE. Based on the results of RMSE only, RF has the best result because the lower the RMSE, the better the data quality. Even though the MLR has the worst RMSE value, it has the fastest runtime, thanks to the simpler algorithm. The runtime may vary depending on the processing power of the computing devices.

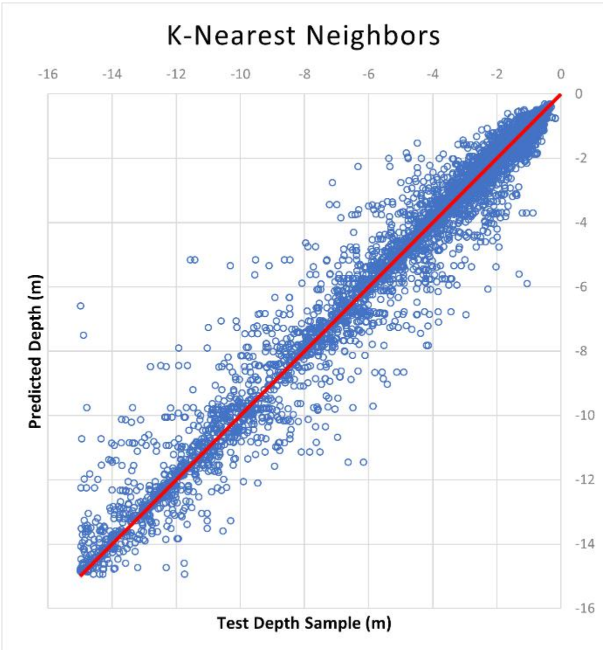


Figure 5. Scatter plot of the test depth sample against predicted depth using KNN.

The RMSE values in **Table 3** were calculated using the test dataset, and then the depth was predicted using the same trained model already used in the previous ML process (i.e., model fitting). The reason for using a test dataset to validate the predicted depth is to create independent data as a control.

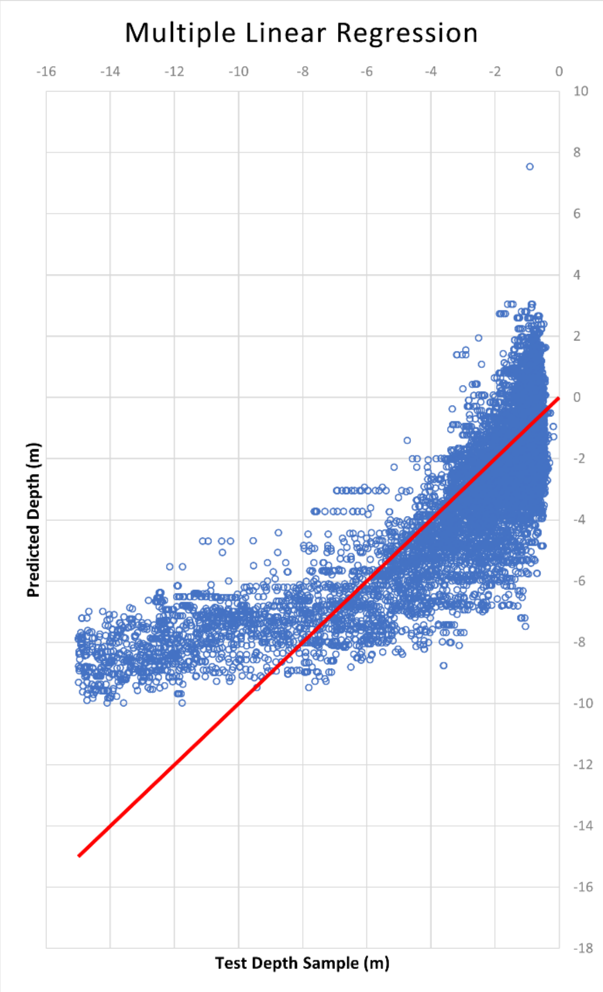


Figure 6. Scatter plot of test depth sample against predicted depth using MLR

Using the predicted depth from the test dataset makes it easier to visualize the relation between the predicted depth and the test dataset (see **Figure 5**, **Figure 6**, and **Figure 7**). Ideally, the test depth samples should equal the predicted depths or y = x (red line in **Figure 5**, **Figure 6**, and **Figure 7**), but that was hardly possible. Thus, the closer a point to the y = x line means the closer it is to the true value. Therefore, by looking into the scatter plots, KNN and RF have better results than MLR. **Figure 8** and **Figure 9** shows the derived depth as DEM that was calculated using KNN, MLR, and RF after they were filtered using the median filter. From **Figure 9**b, the DEM as the SDB results from MLR seems to have plenty of null data in the shallow waters. This might be caused by the derived depth using MLR was beyond the range of 0 to 15 m window. In contrast, the DEM products from KNN and RF methods differ greatly in shallow water areas.

Figure 10 shows a cross-section profile of the predicted depth using three methods in selected locations in the research area. From Figure 10, the profile of KNN and RF models have a good agreement, shown by the red and blue lines fitting well in many locations. Meanwhile, the MLR model represented by the green line shifted from the other two models (KNN and RF) in many locations, especially in the deeper area. Moreover, the MLR model has many spikes along the cross profile in Figure 10. This condition was supported by the RMSE values presented in Table 3, at which the MLR yielded the highest RMSE value (1.79 m) compared to RF with 0.46 m and KNN with 0.47 m.

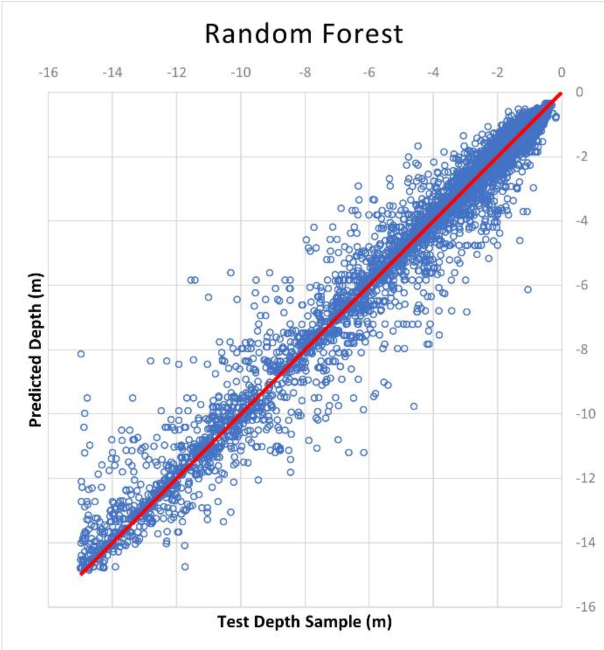
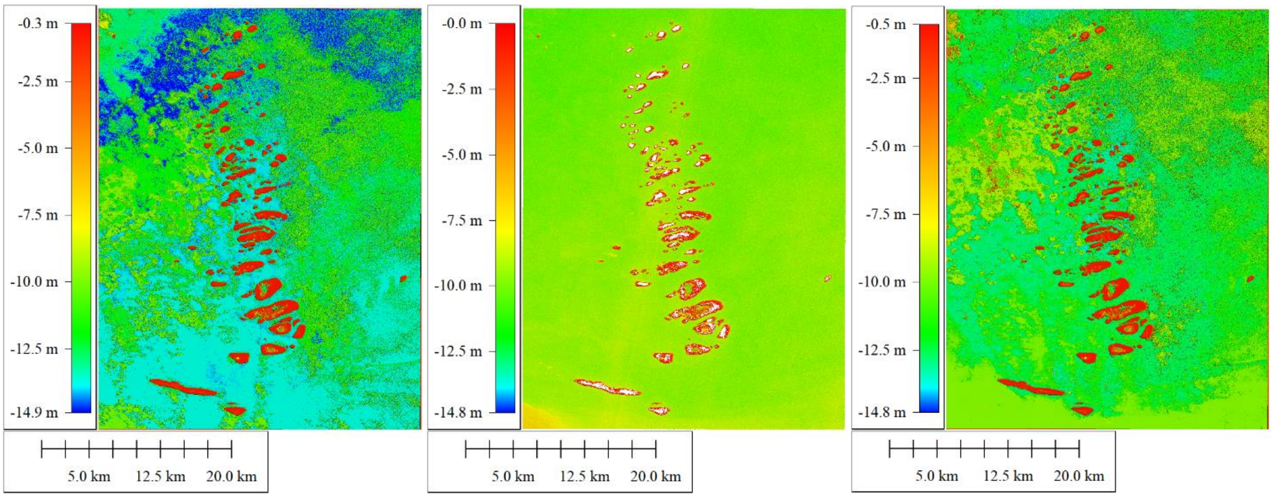


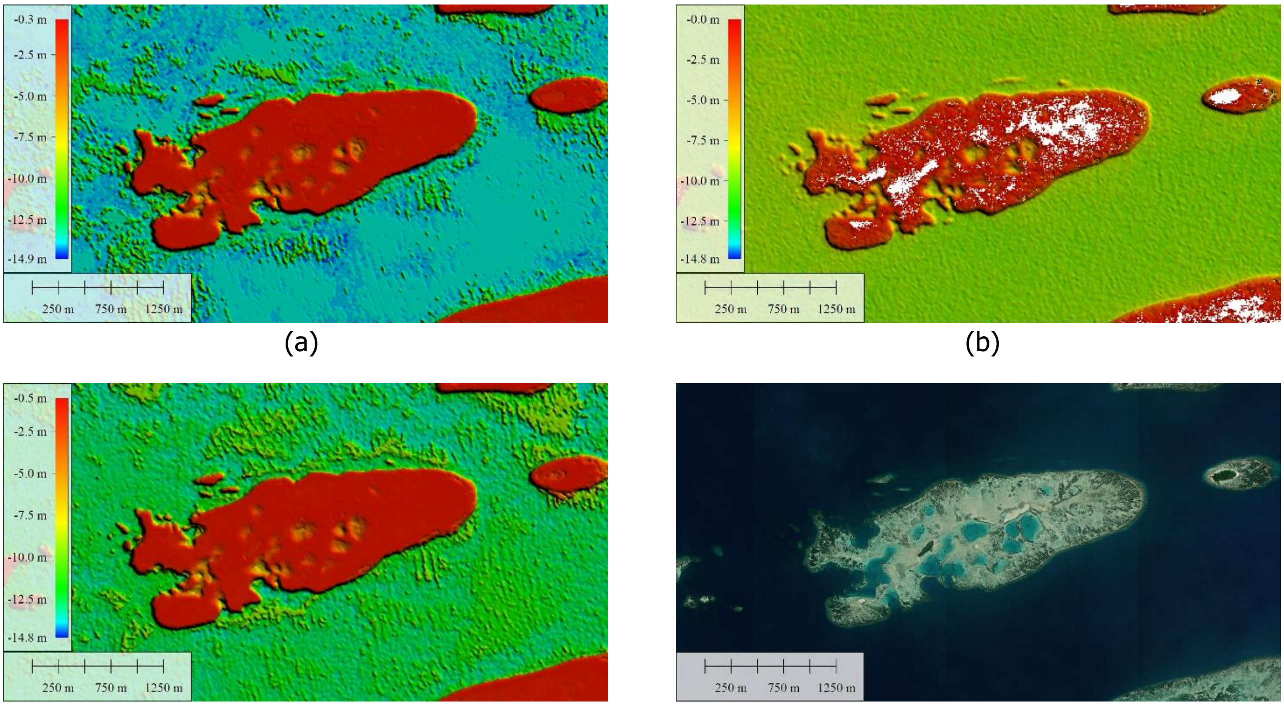
Figure 7. Scatter plot of test depth sample against predicted depth using RF

The predicted depth outside the shallow water areas appears to have values within the range of 10 to 15 m depth, while the typical depth outside the shallow water areas around Kepulauan Seribu, Indonesia, is about 20 to 50 m depth. There are two reasons why this could happen. First, the training dataset only provided 0 to 15 m depth. Second, deriving depth using satellite could only be detected in shallow water areas with a maximum depth of approximately 20 to 30 m.



(a) (b) (c)

Figure 8. Overall view of the predicted depths as DEM that was calculated using (a) KNN, (b) MLR, and (c) RF



(c) (d)

Figure 9. A visualization of the predicted depth as DEM that was calculated using (a) KNN, (b) MLR, and (c) RF. While (d) is satellite image in the similar location as the DEM clip (image from world imagery in Global Mapper software)

Figure 11 shows three cross-section profiles of the predicted depth from different prediction methods outside a shallow water area. All three methods show an almost flat seabed with a few spikes. While a flat seabed is possible, the fact that the real depth could be 20 to 50 m indicates that the predicted depth outside the shallow waters was inaccurate. Therefore, most of the predicted data outside the shallow water areas were removed from the results. However, some derived depths close to the shallow water areas might be accurate because there were slopes that go from shallow to deeper waters. Unfortunately, we could not determine the boundary of which an accurately derived depth close to the shallow water areas except using conventional bathymetric survey data to validate the derived depth.

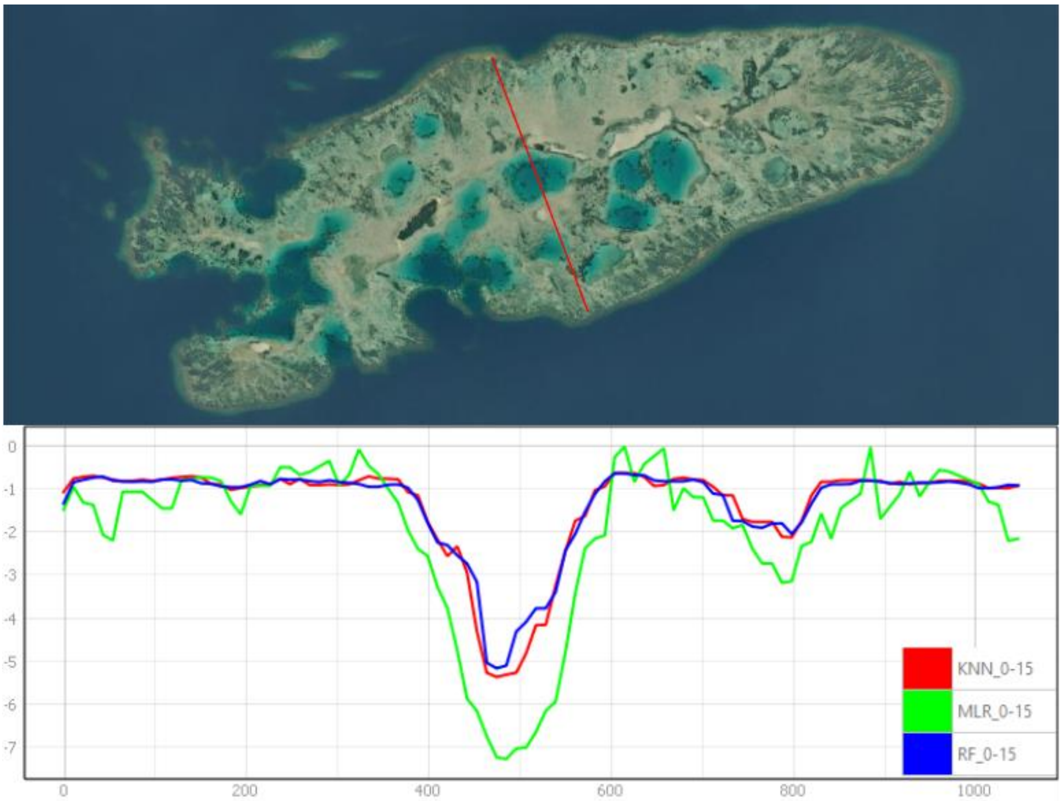


Figure 10. The cross-section profile of depth data of the three ML methods in shallow water areas. Y axis shows depth in m and X axis shows the profile line in m

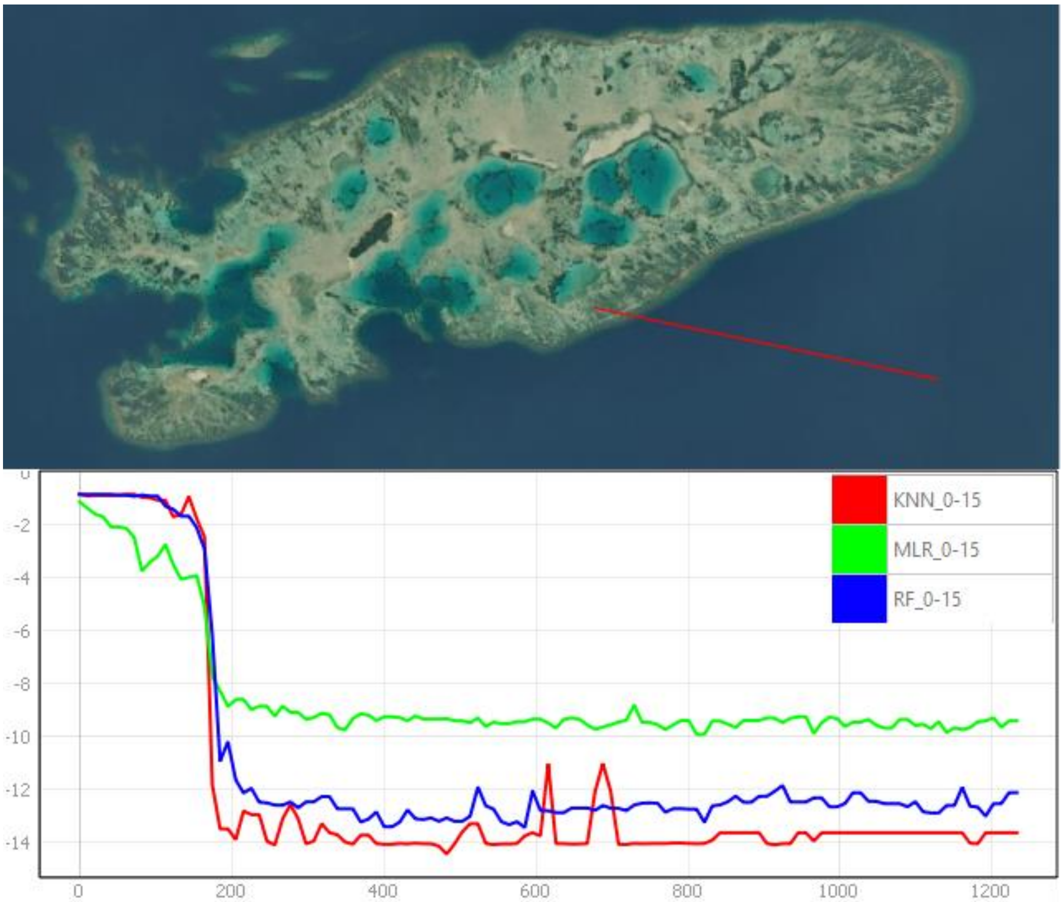


Figure 11. The cross-section profile of depth data of the three ML methods that spans from shallow water area to deeper area. Y axis shows depth in m and X axis shows the profile line in m

Table 4. The maximum and minimum values of the predicted depth from three ML algorithms

ML Method	Max. Value (m)	Min. Value (m)	Range (m)
KNN	-0.34	-14.88	14.54
MLR	0	-14.83	14.83
RF	-0.51	-14.79	14.28

The obtained depth prediction using the three machine learning methods has a different range **Table 4**). The MLR has the widest range of three methods, followed by KNN and RF. The MLR predicted that the depth's range would be wider if the derived depth were not filtered out to 0–15-m depth.

CONCLUSION

Conventional survey vessels have limitations in acquiring depth data in shallow water areas. Thus, research is needed to find an alternative way of obtaining the depth data by using the SDB to fill in the gap of depth data in the shallow water area. The results obtained by this research with RMSE values less than 0.5 m were promising. Using the empirical approach to derive the depth, USV as a platform was used to acquire the depth samples in the shallow water areas to train three ML methods. The depth samples have a 100% confidence level in Order 2 and 95.12% confidence level in Order 1b. That means the depth samples have passed the standard in both orders.

The depth prediction was carried out using 45,141 points of the training data set, or 75% of 60,188 points, and the rest of the depth sample data were used as the test dataset (15,047). Considering the RMSE values, the derived depth using the RF method had the best quality, but its range did not cover to zero depth with the provided training dataset. Only the MLR method provided the derived depth up to zero depth, however, it has the worst RMSE, and some of its derived depth has no data since some locations were outside of 0–15-m depth. On the other hand, KNN has a good balance. It obtained a good RMSE value (which was only 1.3 cm higher than RF) and covered a more depth range than RF. The downside compared to the RF method was it took about 30% longer runtime.

Unless the depth filter after the depth prediction was used, the only usable derived depths were from KNN and RF methods. From these two, RF-derived depth had a better RMSE value (lower than KNN), while KNN was better in the case of the derived depth range.

Nevertheless, these results could be different if some conditions (train and test data, ratio percentage, and depth filter value) and some parameters (each method parameter value) were changed. Strictly speaking, the SBB results are influenced by many factors, such as input data and parameters when performing SDB, atmospheric conditions, and water quality. In addition, this method of deriving depth using ML would not be effective if there are obstacles such as lands or objects above sea level. Therefore, land masking is necessary because any depth data on land is considered false data.

ACKNOWLEDGEMENTS

The authors would like to thank Center for Marine and Coastal Environment of Geospatial Information Agency of Indonesia (Pusat Pemetaan Kelautan dan Lingkungan Pantai, Badan Informasi Geospasial, Indonesia) as the provider of the depth sample data. Because without the depth sample data, this research would not be possible.

REFERENCES

Breiman, L. (2001). Random Forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/A:1010933404324

Caballero, I., & Stumpf, R. (2020). Towards Routine Mapping of Shallow Bathymetry in Environments with Variable Turbidity: Contribution of Sentinel-2A/B Satellites Mission. *Remote Sensing*, *12*(3), 451. https://doi.org/10.3390/rs12030451

Congedo, L. (2021). Semi-Automatic Classification Plugin: A Python tool for the download and processing of remote sensing images in QGIS. *Journal of Open Source Software*, *6*(64), 3172. https://doi.org/10.21105/joss.03172

Delwart, S. (2015). *ESA Standard Document*. (1), 64.

Dewi, R. S., Hartanto, P., Oktaviani, N., Pujawati, I., Nursugi, N., & Aditya, S. (2019). Satellite-derived bathymetry to improve bathymetric map of Indonesia. In T. D. Pham, K. D. Kanniah, K. Arai, G. J. P. Perez, Y. Setiawan, L. B. Prasetyo, & Y. Murayama (Eds.), *Sixth International Symposium on LAPAN-IPB Satellite* (p. 31). Bogor, Indonesia: SPIE. https://doi.org/10.1117/12.2540779

Duplančić Leder, T., Leder, N., & Peroš, J. (2019). Satellite Derived Bathymetry Survey Method—Example of Hramina Bay. *Transactions on Maritime Science*, *8*(1), 99–108. https://doi.org/10.7225/toms.v08.n01.010

Gao, J. (2009). Bathymetric mapping by means of remote sensing: Methods, accuracy and limitations. *Progress in Physical Geography: Earth and Environment*, *33*(1), 103–116. https://doi.org/10.1177/0309133309105657

Gillies, S. (2013). *Rasterio: Geospatial raster I/O for Python programmers*. Mapbox. Retrieved from https://github.com/rasterio/rasterio

Hamylton, S., Hedley, J., & Beaman, R. (2015). Derivation of High-Resolution Bathymetry from Multispectral Satellite Imagery: A Comparison of Empirical and Optimisation Methods through Geographical Error Analysis. *Remote Sensing*, *7*(12), 16257–16273. https://doi.org/10.3390/rs71215829

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., … Oliphant, T. E. (2020). Array programming with NumPy. *Nature*, *585*(7825), 357–362. https://doi.org/10.1038/s41586-020-2649-2

Harrys, R. M. (2022). *Satellite Derived Bathymetry (SDB) GUI* [Python]. Retrieved from https://github.com/rifqiharrys/sdb_gui

International Hydrographic Organization. (2020). *Standards for Hydrographic Surveys S-44* (6th ed.). Monaco: International Hydrographic Organization. Retrieved from https://iho.int/uploads/user/pubs/standards/s-44/S-44_Edition_6.0.0_EN.pdf

Jordahl, K., Bossche, J. V. den, Fleischmann, M., Wasserman, J., McBride, J., Gerard, J., … Leblanc, F. (2020). *geopandas/geopandas: V0.8.1*. Zenodo. https://doi.org/10.5281/zenodo.3946761

Kramer, O. (2013). K-Nearest Neighbors. In O. Kramer, *Dimensionality Reduction with Unsupervised Nearest Neighbors* (pp. 13–23). Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-38652-7_2

Marks, K. (2018). *IHO-IOC GEBCO Cook Book—2018 Progress Report* (p. 429). Maryland, USA: NOAA Laboratory for Satellite Altimetry, College Park.

McKinney, W. (2010). *Data Structures for Statistical Computing in Python*. 56–61. Austin, Texas. https://doi.org/10.25080/Majora-92bf1922-00a

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Cournapeau, D. (2011). Scikit-learn: Machine Learning in Python. *MACHINE LEARNING IN PYTHON*, 6.

Rossi, L., Mammi, I., & Pelliccia, F. (2020). UAV-Derived Multispectral Bathymetry. *Remote Sensing*, *12*(23), 3897. https://doi.org/10.3390/rs12233897

Sagawa, T., Yamashita, Y., Okumura, T., & Yamanokuchi, T. (2019). Satellite Derived Bathymetry Using Machine Learning and Multi-Temporal Satellite Images. *Remote Sensing*, *11*(10), 1155. https://doi.org/10.3390/rs11101155

Tonion, F., Pirotti, F., Faina, G., & Paltrinieri, D. (2020). A Machine Learning Approach to Multispectral Satellite Derived Bathymetry. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, *V-3–2020*, 565–570. https://doi.org/10.5194/isprs-annals-V-3-2020-565-2020

Tranmer, M., Murphy, J., Elliot, M., & Pampaka, M. (2020). *Multiple Linear Regression (2nd Edition)*. 59.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., … Vázquez-Baeza, Y. (2020). SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods*, *17*(3), 261–272. https://doi.org/10.1038/s41592-019-0686-2

Zhu, Y., & Huang, C. (2012). An Improved Median Filtering Algorithm for Image Noise Reduction. *Physics Procedia*, *25*, 609–616. https://doi.org/10.1016/j.phpro.2012.03.133