# Spatio-Temporal Embedding in Deep Learning for Algal Bloom Forecasting

**Kerem Can Bayraktar**

**Supervisor(s): Dr. Jan van Gemert, Attila Lengyel and Robert-Jan Bruintjes**

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
January 29, 2023

Name of the student: Kerem Can Bayraktar
Final project course: CSE3000 Research Project
Thesis committee: Dr. Jan van Gemert, Attila Lengyel and Robert-Jan Bruintjes, Prof. Dr. K.G. Langendoen

## Abstract

The term ”Algal Bloom” refers to the accumulation of algae in a confined geological space. They may harm human health and negatively affect ecological systems around the area. Thus, forecasting algal blooms could mitigate the environmental and socio-economical damages. Particularly, the use of deep learning methods could distinguish underlying patterns such as spatio-temporal dependencies in the available remote sensing data of environmental factors that might cause algal blooms, such as change in water temperature. This research paper will aim to answer the following research question: *Does the inclusion of explicit spatio-temporal embedding methods display a significant improvement for predicting algal blooms?*. The paper will use the UNet architecture and further encodes spatial and temporal information to be explicitly included as features in the training and validation process of deep learning models. The results of the experiment show that the inclusion of explicit spatio-temporal information separately into the feature space exhibits a small increase in performance. However, the combination of spatio-temporal information does not display a significant improvement for the predictions.

## 1 Introduction

The term ”Algal Bloom” refers to the accumulation of algae in a confined geological space, which can be caused by various factors such as increase in temperature or excessive nutrient loads in water sources [1]. A sudden increase in concentration of chlorophyll-a (used by algae in oxygenic photosynthesis) in fresh water resources may further lead to ”Harmful Algal Blooms” (HABs). HABs not only release toxic substances that may harm human health, but they also can affect nearby ecological systems negatively and may threaten the balance of this local ecosystem [2]. They are estimated to cause billions of dollars in damage worldwide [3]. Therefore, forecasting algal blooms in fresh water resources accurately might help in mitigating the environmental and subsequent socio-economical damages they are causing.

The forecasting problem is modeled through the processing of the relevant region’s collected data with varying modalities.The objective of the forecast is to use the *”window size”* number of sequential geo-spatial samples to predict the chlorophyll-a concentration *”prediction horizon”* days ahead. Since in-situ measurements may not be practicable for all regions and all sources of data because of physical inaccessibility or high cost of finances, ”Remote Sensing” is a viable alternative to in-situ measurements. Remote Sensing is the identification of earth’s surface characteristics and the estimation process of their respective properties, referring to the air or space-born remote sensors that are used to gather geological data [4]. We show the visual demonstration of the remote sensing data used for algal bloom forecasting in Figure 1.

The abundance of data gathered through remote sensing, as well as the differing media types (e.g. satellite imagery, meteorological data), make it difficult to combine the gathered data in a meaningful manner for forecasting. The data is traditionally combined manually through the collection of maritime data, which is later modeled either through explicit programming or shallow machine learning shore-based prediction systems [5]. Such methods can lead to material and financial bottlenecks for extensive monitoring, also requiring more man-hours for feature extraction and modeling [6]. As a solution to overhead costs, deep learning can prove to be useful by making use of its advantage of automatic feature learning, which allows the model to identify underlying patterns among features even in noisy and disorganized data. Furthermore, since the data depends on each other considering the varying spatial and temporal factors, modifying state-of-the-art deep learning methods in accordance with the spatio-temporal knowledge may improve the overall performance [7].

Although the use of deep learning methods could improve the performance, the model may still miss relevant information such as the addition of sample geo-locations. Thus, the explicit use of spatial information might be further beneficial as integrating location information into the training process will convey the same domain-specific patterns present in geographical data such as spatial dependencies [8]. In addition, algal bloom forecasting is a time series problem where data is successively collected over a time interval. The collected data includes periodic temporal patterns (e.g. yearly/seasonal patterns) as well as local extrema (e.g. weekly trends, incidental factors). Therefore, the addition of temporal information into the training process is also particularly relevant concerning the spatio-temporal dependencies of algae colonies.

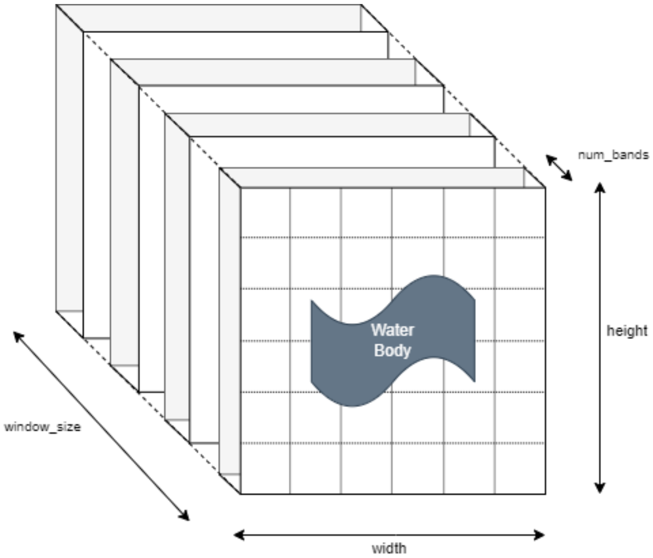

Figure 1: The ”data cube”, an example demonstration of the processed samples within a batch for the algal bloom forecasting problem. A specified number of data band measurements are stacked on top per sample. If batch size is bigger than 1, these cubes are stacked once more on another dimension to be ready for training purposes.

The objective of this paper is to answer the following research question: *Does the inclusion of explicit spatio-temporal embedding methods display a significant improvement for predicting algal blooms*? This paper will use deep learning to forecast the HABs in one of the three fresh water sources in Uruguay with their respective data provided by the research supervisor. Furthermore, it will particularly focus on embedding spatio-temporal information through:

- using a modified convolutional neural network architecture to capture the spatial patterns,
- including spatial data explicitly as a feature to train on to observe further improvements,
- including temporal data to enrich the feature space of training data to capture patterns in the repeating nature of time.

We show that adding spatio-temporal information to remote sensing data slightly improves the accuracy of a CNN for algal bloom forecasting. More importantly, we also show that such addition increases the confidence of class predictions by reducing false positives for the higher chlorophyll-a concentrations.

## 2 Related Work

This section will focus on the relevancy of deep learning with spatio-temporal embedding in comparison with the existing work on algal bloom forecasting.

### 2.1 Algal Bloom Forecasting

Traditional machine learning methods have been previously used for various ecological forecasting purposes. They are suitable for making powerful predictions while excelling in handling sparse data, reducing bias with context-free structure and adequately reflecting the complexity of ecological structures [9]. In addition to traditional ML, artificial neural networks were also successfully applied in predicting algae trends in varying water bodies [10, 11].

### 2.2 Choosing the Neural Network Architecture

Several neural network architectures can improve on the performance with their inclusion of spatial and temporal metadata. Convolutional Neural Networks (CNN) are generally used for image classification and segmentation tasks with convolutional and fully-connected layers that convolve over the input image. This allows the neural network to create ”feature maps”, meaning that it not only includes the weight of the variables for the learning process but also their respective spatial information [12], implicitly providing a richer feature space to the training process. On top of their ability to gather spatial metadata, CNNs are also applicable on time-series data since they are able to derive features from grid-like data within a certain spatial context [13]. Recurrent Neural Networks (RNN) on the other hand are directly suitable for time-series as they can preserve the relevant information from past internal states to look for patterns in the input [14]. Certain variations of RNNs such as Long Short-Term Memory (LSTM) [15] can be particularly effective for making use of information gathered from earlier feed-back signals. Additionally, certain combinations of aforementioned structures could improve performance even further. Examples include Conv-LSTM [16] which is a mixture of convolutional layers in a RNN structure, or or UNet [17], a modified CNN that achieves more precise analysis through fewer training data. This way, the network can preserve both spatial and temporal metadata to create context. Therefore the conducted experiment will make use of the UNet structure because of its scalability for large datasets as a CNN, while still maintaining a simple structure to implement with a previously established performance in image segmentation.

### 2.3 Explicit Use of Spatio-temporal Information

Recently, neural networks were observed to be more suitable than traditional machine learning approaches for the purpose of spatio-temporal forecasting [18], ranging from forecasting precipitation [19] to photo-synthetically active radiation [20]. In its essence, forecasting is a multivariate classification/regression problem and neural networks may reveal patterns that are not easily observable in large amount of data. However, space and time based dependencies are often overlooked for tasks such as determining the type of land cover [21] or short and long-term patterns for traffic flow [22].

Although neural network architectures may include meta information about space and time for implicit use, there were no explicit spatio-temporal embedding example as a feature in the observed papers. Therefore it's not clear what effect it would have on the performance of the model. This paper will focus on explicit inclusion of spatio-temporal data with their respective encoding as features.

## 3 Dataset

This section will focus on the specific dataset provided by the project supervisor with the help of domain experts. It will provide some initial analysis of the data solely through the **palmar** water reservoir to provide a clearer understanding of the data distribution. It will also explain how it lead to design choices later on through the experiment setup.

### 3.1 Processing the Raw Data

Relevant data could be categorised under three main properties: biological, meteorological and surface water temperature. The collected data mainly refers to three regions and respective water bodies nearby in Uruguay: palmar, bonete and baygorria. Biological data was acquired through the Ministry of Environment of Uruguay and consists of chlorophyll-a concentration, water turbidity and colored dissolved organic matter (CDOM) concentration. Meteorological data was gathered through NOAA Global Forecast System and consists of the following bands: air temperature, cloud coverage, precipitation, radiation, relative humidity, u wind and v wind. Finally, surface water temperature was acquired through satellites “AQUA” and “TERRA” through MODIS of NASA.

Files are read from TIFF/GeoTIFF format and subsequently combined through the use of Rasterio, a Python library that allows read and write operations for GeoTIFF files, to comply with the conventional geospatial data format GDAL. Therefore, even after the combination of aforementioned bands, the spatial and temporal metadata could be preserved for further use in data preprocessing for model training later on. As a result, all the findings could be projected onto an abstracted geological data structure on which the data loader could work on loading in batches for training the model.

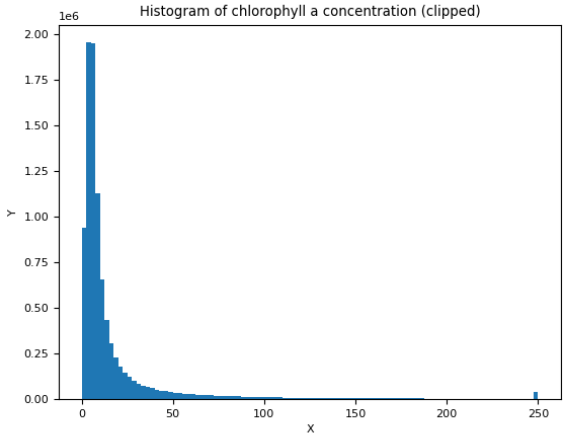

Figure 2: An overview of how chlorophyll-a values are distributed in the clipped range of [0, 250].

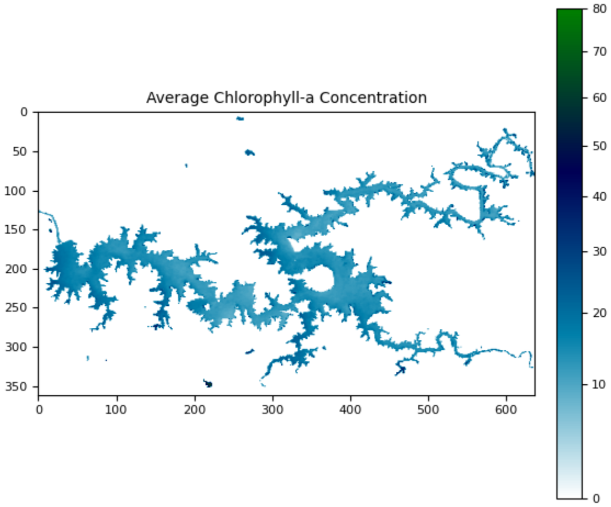

Figure 3: An example of the spatial distribution of chlorophyll-a concentration averaged per pixel over 365 samples from 2016-09-23 to 2022-06-19.

### 3.2 Chlorophyll-a Analysis

- **Sparse and imbalanced data.** Within the given palmar dataset, 77.926% of the water-pixels have a measurement over the time interval of late 2016 to mid-2022. As can be seen from Figure 2, the provided chlorophyll-a data is disproportionately distributed. 80 µg/L of chlorophyll-a concentration is considered high by the domain expert, which implies that only 4.885% of the non-NaN data is above the threshold. Upon the project supervisor's suggestion, the binning range was chosen as [0-10, 10-30, 30-75, 75+] µg/l. For this given range the distribution of the data is as follows: 63.574%, 23.185%, 7.963%, 5.273%. The input and predicted values will fall into one of these bins for classification purposes.

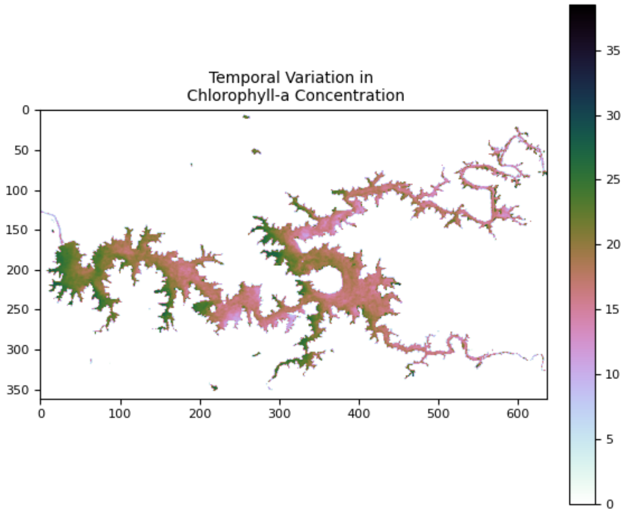

Figure 4: Overview of the standard distribution/the fluctuation per pixel over 365 samples.

  Artificial oversampling of minority classes, undersampling of the majority class and weighted loss methods were considered possible options to tackle the imbalanced data problem. Over/undersampling methods might interfere with the underlying spatio-temporal patterns of the sequential environmental data, considering that the data distribution will change without the "natural randomness". For that reason, the handling of the imbalanced data was done through class-balanced loss [23], which is a weighted loss approach that gives class weights by the inverse of their percentage within the data.

- **NaN values.** NaN values in chlorophyll-a values refer to invalid data readings from remote sensors. The lack of available data is either caused by the cloud coverage since the readings are remotely collected through spectrophotometer, or areas that fall outside the water reservoir and hence don't have readings for light reflecting from algae in the water. The percentages of pixels with a non-NaN chlorophyll-a readings are 0.112%, which shows that there's a need for handling NaN values later in preprocessing.

- **Spatio-temporal distribution.** As can be seen from Figure 3 and 4, high concentrations of chlorophyll-a are mostly accumulated over the coastal boundaries of water bodies with relatively high fluctuation with discrete boundaries. Encoding spatial information and using it in the feature space might help with finding such border patterns to have an effect on the prediction. Moreover, Figure 5 shows how the chlorophyll-a readings change through temporal samples over time. Therefore, encoding time and including it as a feature band might help the model to catch these patterns to make more precise predictions.

## 4 Integrating Spatio-Temporal Information

This section will explain the chosen methods to include explicit spatio-temporal information within the training and validation process. It will further provide reasonings for methodological and design choices.

### 4.1 Model Architecture

Because of its ability to perform better with fewer training images while still benefiting from its modified convolutional neural network architecture, UNet was chosen as the model. An overview of its architecture can be seen in Figure 6. PyTorch Lightning library was utilized to wrap the model in order to process the incoming batches before continuing further with the training of the data. A demonstration of the output of batch processing (and what the model receives) can be seen in Figure 1. The data cube has values for respective bands placed on the 2-dimensional grid space and bundled together per day, which is then combined further with previous days' stacked bands to create a 4-dimensional input tensor. Subsequently, the UNet model applies convolution operations over these tensors to deduce the feature mapping.

### 4.2 Spatio-temporal Encoding

The sampling for training and validation is done through ”Bounding Boxes”. Bounding Boxes within this context are structures used as a helping tool for fetching image data from the dataset in memory. Because metadata such as timestamp and coordinate values are not directly accessible through tensor indexing, Bounding Boxes allow the creation of spatio-temporal ”Region of Interests” for the sampler. It composes of six parameters including the maximum and minimum x,y coordinates of the sampled region as well as the minimum and maximum timestamps among the samples. Then, the sampler will fetch the data from the dataset in accordance with the region of interest set by the Bounding Box.

Since all of the bands are separate images, the query returns every available band per day and stacks them in a dimension to be returned as input images to the model seen in Figure 1. At this stage, the accessed timestamps per each sample are encoded into the day of the year and added as a new ”pseudo-band”, clipped and scaled, and subsequently stacked with the rest. Unlike the timestamps which change per each sample, spatial information is preserved among the samples within the same batch because of the way the sampler functions with the BoundingBoxes. Therefore, it is sufficient to access the coordinate information for the corners of the bounding box in order to create 2-D coordinate arrays. All of the parameters are inferred by the data loader from the processed dataset. The spatial information is then extracted through the sampler's region of interest, while the temporal information is gathered through an iteration of the samples to be matched with their respective timestamps.

## 5 Experiments

The experiment was carried out in Python and Torch library. Main packages that were used are PyTorch[1] for the model, TorchGeo[2] for handling geospatial data, and PyTorch Lightning[3] for increased model readability and logging. The experiment was conducted on DelftBlue Supercomputer [24].

### 5.1 Setup

**Loading the Processed Data**

The purpose of the data loader is to format the input .tiff files into tensors[4] for conducting the deep learning steps after the data is cleaned and processed into one file. The input data is first converted into a PyTorch Dataset in a tuple with the format $(image, mask, target)$. Image refers to the batch of stacked samples, and has the shape of $(batch\_size, window\_size, num\_bands, input\_size, input\_size)$. Mask is the respective mask for some of the data bands such as the biological data in order to reduce the use of computational resources for irrelevant spatial information by masking the raster dataset. Target refers to the target sample that the model is trying to predict values for the respective bands with the shape of $(batch\_size, num\_bands, input\_size, input\_size)$.

After the data loader is set, the data were loaded to tensors. However, as the entire dataset includes enormous amounts of samples with different bands, loading the full dataset for training purposes is not ideal. Therefore, the sampler has the objective of (in this case, randomly) choosing instances from all the samples for training the model with a reasonable amount of computational power. The same sampling process can be repeated for testing purposes while making sure that the chosen sample is within the region of interest. The chlorophyll-a concentration (µg/L) data was chosen as the ground truth for the model to compare its predictions.

**Data Pre-processing**

After the data is received by the model, the following operations were applied:

- Extreme values were clipped from the input tensor to fit in the range of [$10^{-6}$, clip_value]. Likewise, the ground truth was also clipped to fit in the range of [0, clip_value]. In addition, biological data bands and the precipitation band went through Yeo-Johnson Transformation [25] in order to make the data more normal distribution-like.
- NaN values for the input tensor were replaced with mean values per band to minimize their effect on normalization. The ground truth were replaced with 0 to bin them

[1] https://pytorch.org/

[2] https://torchgeo.readthedocs.io/en/stable/

[3] https://www.pytorchlightning.ai/

[4] https://pytorch.org/docs/stable/tensors.html

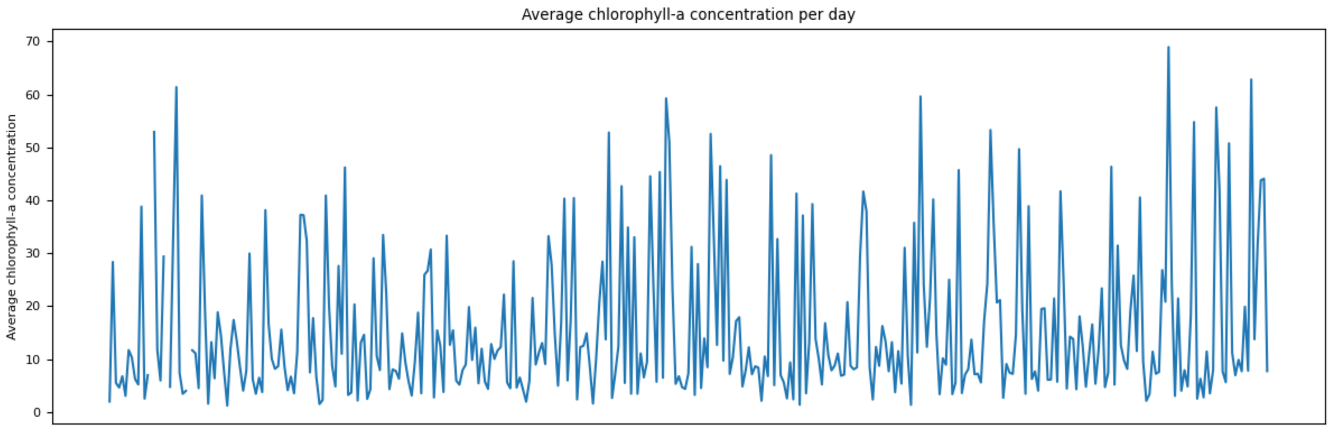

Figure 5: An overview of time-series data for chlorophyll-a throughout the time interval. Periodic peaks and troughs could be observed and are relevant for training data considering the imbalanced distribution of the data.

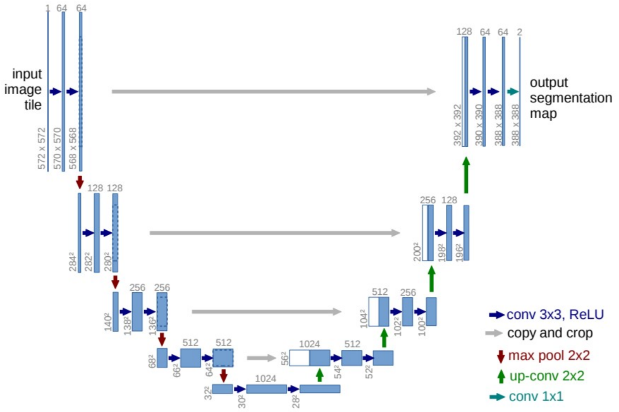

Figure 6: The architecture of the UNet Model [17]. The model first applies convolution and max pooling to create feature maps and then scales upwards with the same mapping to minimize distortion.

into the first label index. The replaced zeros will be ignored later on for loss and accuracy calculations while carrying out the training and test. Even though the model will predict values for it, it will not ”learn” from these instances.

- The mean and standard deviation for each of the included bands were calculated. Then, the input tensor was normalized through standard score (Z-score) calculation with the assumption that it’s normally distributed. Since time band is not normally distributed, it was scaled into the range of [0,1] instead.
- Ground truth (labeled images) tensor was bucketized into class boundaries for classification purposes, with the first index indicating previously NaN or negative chlorophyll-a readings.

After the pre-processing was done, the window size and band dimensions were collapsed into a single dimension in order to feed the UNet with $window\_size * num\_bands$ input channels. As a final step, spatial information that was received from batch processing was stacked on top of the collapsed dimensions to finalize the spatio-temporal data for training.

Since the UNet architecture preserves the number of channels between input and output, the predictions could be directly compared with the labeled images for loss and accuracy calculations. The model’s output predictions were finally converted into probabilities through the softmax function for plotting class probabilities. The UNet model implementation was a modified version of a PyTorch UNet implementation for image segmentation [5].

## 5.2 Results

### Parameters

The prediction horizon of 1 and window size of 1 were chosen as parameters for the forecasting problem. The training was optimized through Adam optimizer with a learning rate of 0.0001 and a weight decay of 0.00001 after manual optimization. A batch size of 16 and 1000 training samples per epoch were used for loading and sampling of the data. All 11 bands were used as part of the training and the predictions were placed in one of 4 classes, of which the binning range was chosen [0-10, 10-30, 30-75, 75+] by the domain expert. The experiment was conducted with data from the ”palmar” reservoir. Cross-entropy loss with initial weights as inverse percentage of class distribution was chosen in order to tackle the imbalanced state of the data. The data that is present before the date 31 December 2021 is used for training purposes and the model is then validated on unseen data, which includes the last 6 months’ data until 19 June 2022.

### Loss and Accuracy

Figure 7 shows confusion matrices of each embedding type averaged over 20 epoch normalized per row for visualizing the distribution of the predictions, with labels referring to indexes of the designated bins.

The accuracy and weighted cross entropy loss values of respective embedding types can be seen in Table 1. The separate inclusion of spatial and temporal embeddings display a

[5] https://github.com/milesial/Pytorch-UNet

| embed_type | train_acc | train_loss | val_acc | val_loss |
|---|---|---|---|---|
| none | 0.7668 | 1.095 | 0.3336 | 1.515 |
| spatial | 0.7279 | 0.8591 | 0.3736 | 1.467 |
| temporal | 0.7384 | 0.9516 | 0.375 | 1.418 |
| spatio-temporal | 0.7759 | 0.8707 | 0.3305 | 1.469 |

Table 1: Table to test display accuracy and loss scores.

decrease in validation losses as well as a slight improvement in validation accuracies.The single instances of spatial and temporal embeddings have better prediction percentages for intervals 10-30 and 30-75 than the version without embeddings.

The combined spatio-temporal embedding shows a decrease in loss although it does not show an improvement in accuracy compared to the one without any embeddings. The confusion matrix of the experiment without any embeddings display the highest percentage for predicting 0-10 and 75+ intervals, while spatio-temporal embedding version predicts the intermediate classes with higher percentages.

## 6 Discussions and Conclusion

The confusion matrices show that the inclusion of explicit spatio-temporal data results in a gradual transition for predictions. The confusion matrix of the spatio-temporal embedding instance shows that even though the algorithm predicts with a lower accuracy, it is more likely to choose a closer class in case of a misclassification. Misclassifications between adjacent class labels are with a higher percentage compared to its counterpart without embedding. For example, the percentage of predicting an encountered 0-10 value as a 75+ value is 3 times lower for the spatio-temporal embedded model. This can be visually demonstrated by observing the boxes alongside the diagonal to be of lighter colors as well as darker colors for bottom left and top right. Such a distinction is most likely caused by the cross entropy loss with class percentage as weights, considering that the loss metric is punishing the algorithm for predicting false positives with bigger difference between the bin values. The difference will be more visible for the one including spatio-temporal embedding as the algorithm will shape its predictions to be closer to its spatial and temporal neighbors to avoid bigger loss values in the training. Therefore it's more probable to make misclassifications between pixels that are spatio-temporally connected and predict ”conservatively” rather than predicting high values more often for achieving a higher accuracy score. This approach to predicting might be helpful for gradual increases around the boundaries of water bodies and highlight areas that do display a possibility of blooming, rather than directly spotting areas with high concentrations of chlorophyll-a.

To conclude, the results show that the inclusion of explicit spatial and temporal information by embedding them into the input data exhibits a small increase in performance. The combined addition of spatio-temporal information did not display a significant improvement for the predictions. However, the combination still provides valuable insight by demonstrating how the model can pick up spatio-temporal dependencies even with similar accuracy scores.

## 7 Limitations and Future Work

Certain limitations were encountered through the research process that could have been replaced with alternatives or investigated further for more fitting implementations. Although UNet was deemed as an appropriate model for previous instances [26] of application on algal bloom forecasting, overall accuracy and the loss of the model shows that it was not suited for the data modalities that were used in the experiment. The use of a different model that can detect spatio-temporal dependencies such as Conv-LSTM might be a more promising solution. Furthermore, the data imbalance issue is still present in the training data even though it's attempted to be solved through the loss function. Such an imbalance could be solved by more complex methods such as Two Stage Resampling for CNNs [27] without ruining the underlying data patterns. For improving on the spatio-temporal encoding methods, spatial interpolation [28] could help to catch deep spatial patterns as well as wavelet analysis for time series [29] at pre-processing stage could improve to highlight the time series patterns. Finally, the handling of the NaN values could be explored further to see if there are better alternatives for filling them in as well as avoiding them.

## 8 Responsible Research

The most important ethical concern regarding this paper's responsible contribution to science is the reproducibility of the experiment. The dataset that was provided to the research group was acquired by the project supervisor from Uruguayan researchers. Therefore, not all of the data for the given time and space intervals that is present in the experiment is publicly accessible, limiting the repeatability of the findings. Furthermore, because of internal functions of PyTorch, fully reproducing the results across different platforms and releases is not guaranteed[6]. However, certain steps were taken to minimize the effects of such limitations.

Firstly, a manual seed of 42 was set for controlling the sources of randomness throughout the the model operations such as data loading or convolution operations on CUDA tensors. Secondly, non-deterministic algorithms used in the training process were replaced with their deterministic counterparts whenever they had alternatives by making the Trainer object deterministic. Thirdly, because of the way training and validation data is split, the algorithm is making its predictions always on the same 37 samples that are collected after 31st December 2021. Thus, it will allow a deterministic comparison of the predictions among models with differing parameters. Finally, the code that includes the implementation details of the experiment will be hosted on TU Delft's repository in case the data becomes publicly accessible at a later stage.

## References

[1] A. Mozo *et al.*, “Chlorophyll soft-sensor based on machine learning models for algal bloom predictions,” *Scientific Reports*, vol. 12, no. 1, 2022. DOI: 10.1038/s41598-022-17299-5.

[6] https://pytorch.org/docs/stable/notes/randomness.html

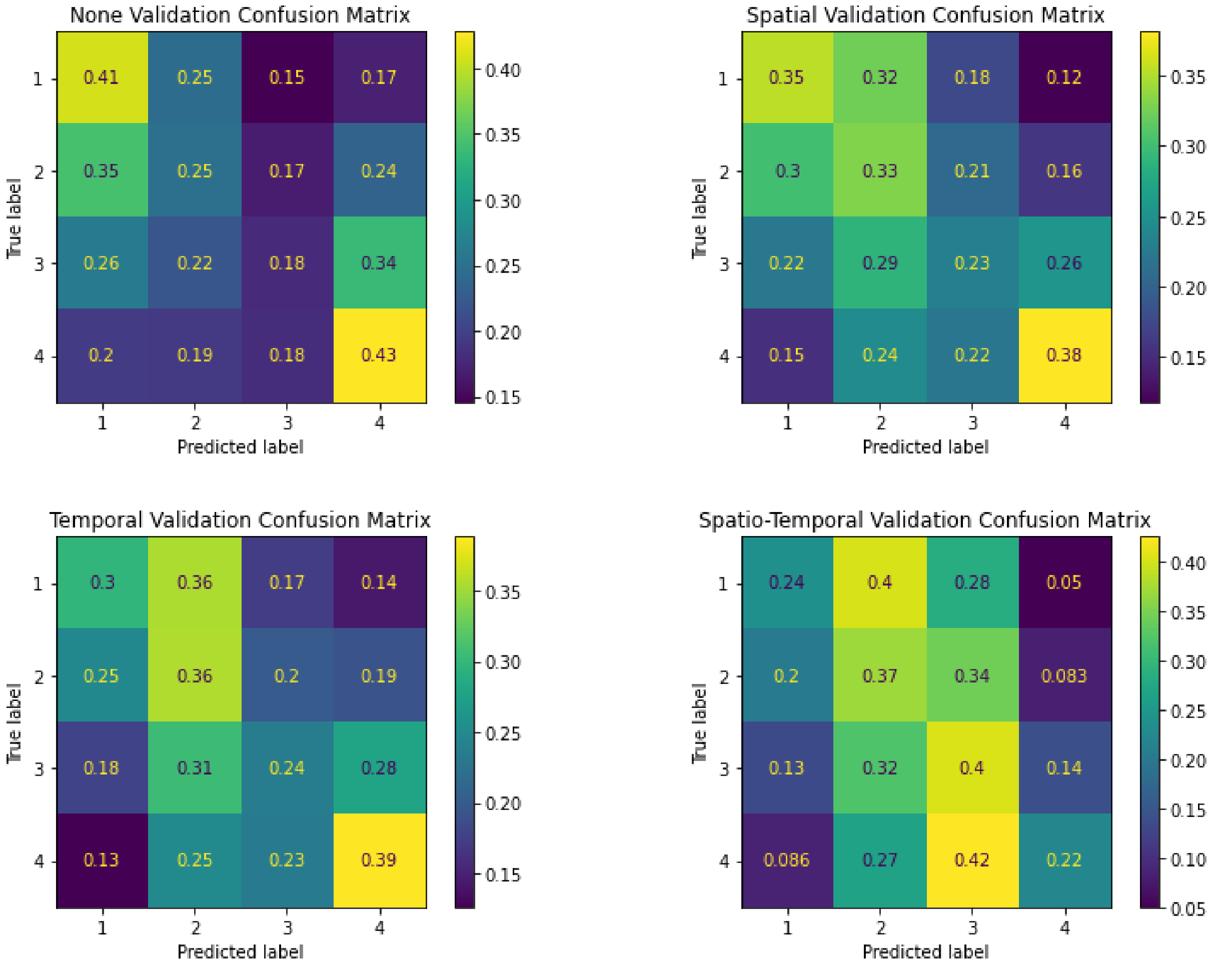

Figure 7: Comparison of confusion matrices of varying embedding types, averaged over 20 epochs and normalized per row.

[2] J. Heisler *et al.*, "Eutrophication and harmful algal blooms: A scientific consensus," *Harmful Algae*, vol. 8, no. 1, pp. 3–13, 2008. DOI: 10.1016/j.hal.2008.08.006.

[3] J. Wen, J. Yang, Y. Li, and L. Gao, "Harmful algal bloom warning based on machine learning in maritime site monitoring," *Knowledge-Based Systems*, vol. 245, p. 108 569, Jun. 2022. DOI: 10.1016/j.knosys.2022.108569.

[4] *What is remote sensing and what is it used for?* [Online]. Available: https://www.usgs.gov/faqs/what-remote-sensing-and-what-it-used.

[5] P. M. Glibert, "Eutrophication, harmful algae and biodiversity — challenging paradigms in a world of complex nutrient changes," *Marine Pollution Bulletin*, vol. 124, no. 2, pp. 591–606, Nov. 2017. DOI: 10.1016/j.marpolbul.2017.04.027.

[6] C. Janiesch, P. Zschech, and K. Heinrich, "Machine learning and deep learning," *Electronic Markets*, vol. 31, no. 3, pp. 685–695, 2021. DOI: 10.1007/s12525-021-00475-2.

[7] F. Amato, F. Guignard, S. Robert, and M. Kanevski, "A novel framework for spatio-temporal prediction of environmental data using deep learning," *Scientific Reports*, vol. 10, no. 1, Dec. 2020. DOI: 10.1038/s41598-020-79148-7.

[8] P. Liu and F. Biljecki, "A review of spatially-explicit geoai applications in urban geography," *International Journal of Applied Earth Observation and Geoinformation*, vol. 112, p. 102 936, Aug. 2022. DOI: 10.1016/j.jag.2022.102936.

[9] F. Huettmann *et al.*, "Use of machine learning (ml) for predicting and analyzing ecological and 'presence only' data: An overview of applications and a good outlook," *Machine Learning for Ecology and Sustainable Natural Resource Management*, pp. 27–61, Nov. 2018. DOI: 10.1007/978-3-319-96978-7_2. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-96978-7_2.

[10] P. Coad, B. Cathers, J. E. Ball, and R. Kadluczka, "Proactive management of estuarine algal blooms using an automated monitoring buoy coupled with an ar-

tificial neural network," *Environmental Modelling & Software*, vol. 61, pp. 393–409, 2014, ISSN: 1364-8152. DOI: https://doi.org/10.1016/j.envsoft.2014.07.011. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1364815214002126.

[11] S. Lee and D. Lee, "Improved prediction of harmful algal blooms in four major south korea's rivers using deep learning models," *International Journal of Environmental Research and Public Health*, vol. 15, no. 7, p. 1322, 2018. DOI: 10.3390/ijerph15071322.

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *Computer Vision – ECCV 2014*, pp. 346–361, 2014. DOI: 10.1007/978-3-319-10578-9_23.

[13] Y. Xu, C. Hu, Q. Wu, Z. Li, S. Jian, and Y. Chen, "Application of temporal convolutional network for flood forecasting," *Hydrology Research*, vol. 52, no. 6, pp. 1455–1468, 2021. DOI: 10.2166/nh.2021.021.

[14] M. Hüsken and P. Stagge, "Recurrent neural networks for time series classification," *Neurocomputing*, vol. 50, pp. 223–235, 2003. DOI: 10.1016/s0925-2312(01)00706-8.

[15] R. C. Staudemeyer and E. R. Morris, "Understanding lstm - a tutorial into long short-term memory recurrent neural networks," *ArXiv*, vol. abs/1909.09586, 2019.

[16] X. Shi, Z. Chen, H. Wang, D. Yeung, W. Wong, and W. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds., 2015, pp. 802–810. [Online]. Available: https://proceedings.neurips.cc/paper/2015/hash/07563a3fe3bbe7e3ba84431ad9d055af-Abstract.html.

[17] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds., Cham: Springer International Publishing, 2015, pp. 234–241, ISBN: 978-3-319-24574-4.

[18] S. Wang, J. Cao, and P. S. Yu, "Deep learning for spatio-temporal data mining: A survey," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 8, pp. 3681–3700, 2022. DOI: 10.1109/tkde.2020.3025580.

[19] A. Akbari Asanjan, T. Yang, K. Hsu, S. Sorooshian, J. Lin, and Q. Peng, "Short-term precipitation forecast based on the persiann system and lstm recurrent neural networks," *Journal of Geophysical Research: Atmospheres*, vol. 123, no. 22, 2018. DOI: 10.1029/2018jd028375.

[20] L. Feng, W. Qin, L. Wang, A. Lin, and M. Zhang, "Comparison of artificial intelligence and physical models for forecasting photosynthetically-active radiation," *Remote Sensing*, vol. 10, no. 11, p. 1855, 2018. DOI: 10.3390/rs10111855.

[21] N. N. Navnath, K. Chandrasekaran, A. Stateczny, V. M. Sundaram, and P. Panneer, "Spatiotemporal assessment of satellite image time series for land cover classification using deep learning techniques: A case study of reunion island, france," *Remote Sensing*, vol. 14, no. 20, p. 5232, 2022. DOI: 10.3390/rs14205232.

[22] R. Asadi, "Deep learning models for spatio-temporal forecasting and analysis," Ph.D. dissertation, University of California Irvine, 2020.

[23] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, and S. Belongie, "Class-balanced loss based on effective number of samples," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019. DOI: 10.1109/cvpr.2019.00949.

[24] D. H. P. C. C. (DHPC), *DelftBlue Supercomputer (Phase 1)*, https://www.tudelft.nl/dhpc/ark:/44463/DelftBluePhase1, 2022.

[25] I.-K. Yeo and R. Johnson, "A new family of power transformations to improve normality or symmetry," *Biometrika*, vol. 87, no. 4, pp. 954–959, 2000. DOI: 10.1093/biomet/87.4.954.

[26] C. Feng, S. Wang, and Z. Li, "Long-term spatial variation of algal blooms extracted using the u-net model from 10 years of goci imagery in the east china sea," *Journal of Environmental Management*, vol. 321, p. 115 966, 2022, ISSN: 0301-4797. DOI: https://doi.org/10.1016/j.jenvman.2022.115966. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0301479722015390.

[27] M. Koziarski, "Two-stage resampling for convolutional neural network training in the imbalanced colorectal cancer image classification," *2021 International Joint Conference on Neural Networks (IJCNN)*, Apr. 2021. DOI: 10.1109/ijcnn52387.2021.9533998.

[28] D. Zhu, X. Cheng, F. Zhang, X. Yao, Y. Gao, and Y. Liu, "Spatial interpolation using conditional generative adversarial neural networks," *International Journal of Geographical Information Science*, vol. 34, no. 4, pp. 735–758, 2020. DOI: 10.1080/13658816.2019.1599122. eprint: https://doi.org/10.1080/13658816.2019.1599122. [Online]. Available: https://doi.org/10.1080/13658816.2019.1599122.

[29] M. Liu, J. He, Y. Huang, T. Tang, J. Hu, and X. Xiao, "Algal bloom forecasting with time-frequency analysis: A hybrid deep learning approach," *Water Research*, vol. 219, p. 118 591, 2022, ISSN: 0043-1354. DOI: https://doi.org/10.1016/j.watres.2022.118591. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0043135422005449.