Team: Wunderpus Octopus

Deniz Olgu Devecioglu, Francesca Balestrieri, Ingrida Semenec, Kshitiz Parihar, Nadir Hajouji, Saswat Mishra

Modeling the relationship between biogeochemical layers and chlorophyll density

TEAM WUNDERPUS OCTOPUS

NEW ATLANTIS

Modeling the relationship between biogeochemical layers and chlorophyll density

Select Ocean Regions:
Shallow regions are significant due to their enhanced light availability, nutrient recycling, and diverse ecosystems, supporting higher primary productivity and biodiversity.

We selected region of interest:

- North Sea

Select time frame : Sep 1997 - Dec 2021

Research questions

- What are the primary biogeochemical and physical factors influencing chlorophyll density in various shallow sea regions?
- How can these factors be quantitatively integrated into a robust predictive model for chlorophyll density?

Combining observation data with simulated data

Copernicus Marine Datasets included

- Dataset 1: Global Ocean Color (satellite observations)
  - Chlorophyll
- Dataset 2: Global Ocean Biochemistry (simulated)
  - O2; NO3; PO4; Si; Fe
- Dataset 3: Global Ocean OSTIA (using in-situ & satellite data)
  - Sea Surface Temperature
- Dataset 4: Multi Observation Global Ocean (using in-situ & satellite data)
  - Sea Surface Salinity and Sea Surface Density
- Dataset 5: Global Ocean Surface Carbon (From model based on in-situ data)
  - Dissolved inorganic carbon; Total alkalinity; Surface partial pressure of CO2 Sea water pH; Calcite saturation state; Aragonite saturation state; Surface downward flux of CO2

XGBoost Modeling Approach

Extract data from Copernicus Mariner Datasets

Data Preprocessing

- Match resolution between datasets
- Keep chlorophyll values within 99 percentile

421801 data points and 19 features

80:20 split

Test set

80:20 split

Training and Validation sets

Remove correlated features

- Baseline model:
  - Chlorophyll prediction = average value of chlorophyll in the training set
- Regression models: XGBoost
  - Bayesian hyperparameter tuning to minimize 5-fold CV RMSE on training set

Evaluation Metric for model comparison: Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE) on the validation set

XGBoost Results

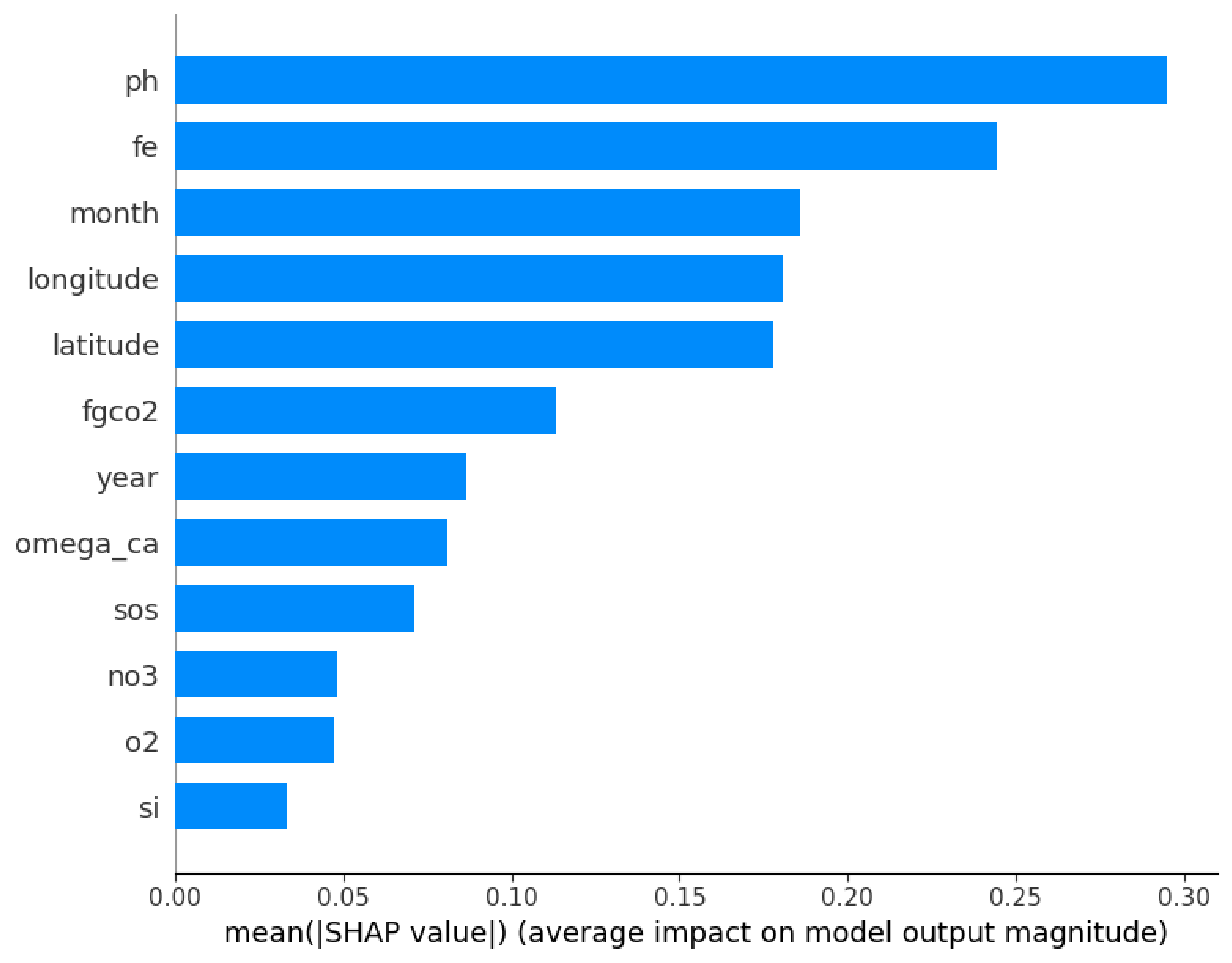
Keep only one of the features among corr > 0.8

Final feature set:
['latitude', 'longitude', 'year, 'month', 'fgco2', 'omega_ca', 'ph', 'fe', 'no3', 'si', 'o2', 'sos']

Model performance on validation set after hyperparameter optimization

Model	RMSE	MAPE
Baseline	1.03	1.08
XGBoost	0.32	0.17

SHAP-based feature importance using XGBoost model on the test set



ConvLSTM

- Goal: Incorporate both time and space into the neural network architecture
- Strategy: combine a CNN (which handles spatial data) and LSTM (for time)

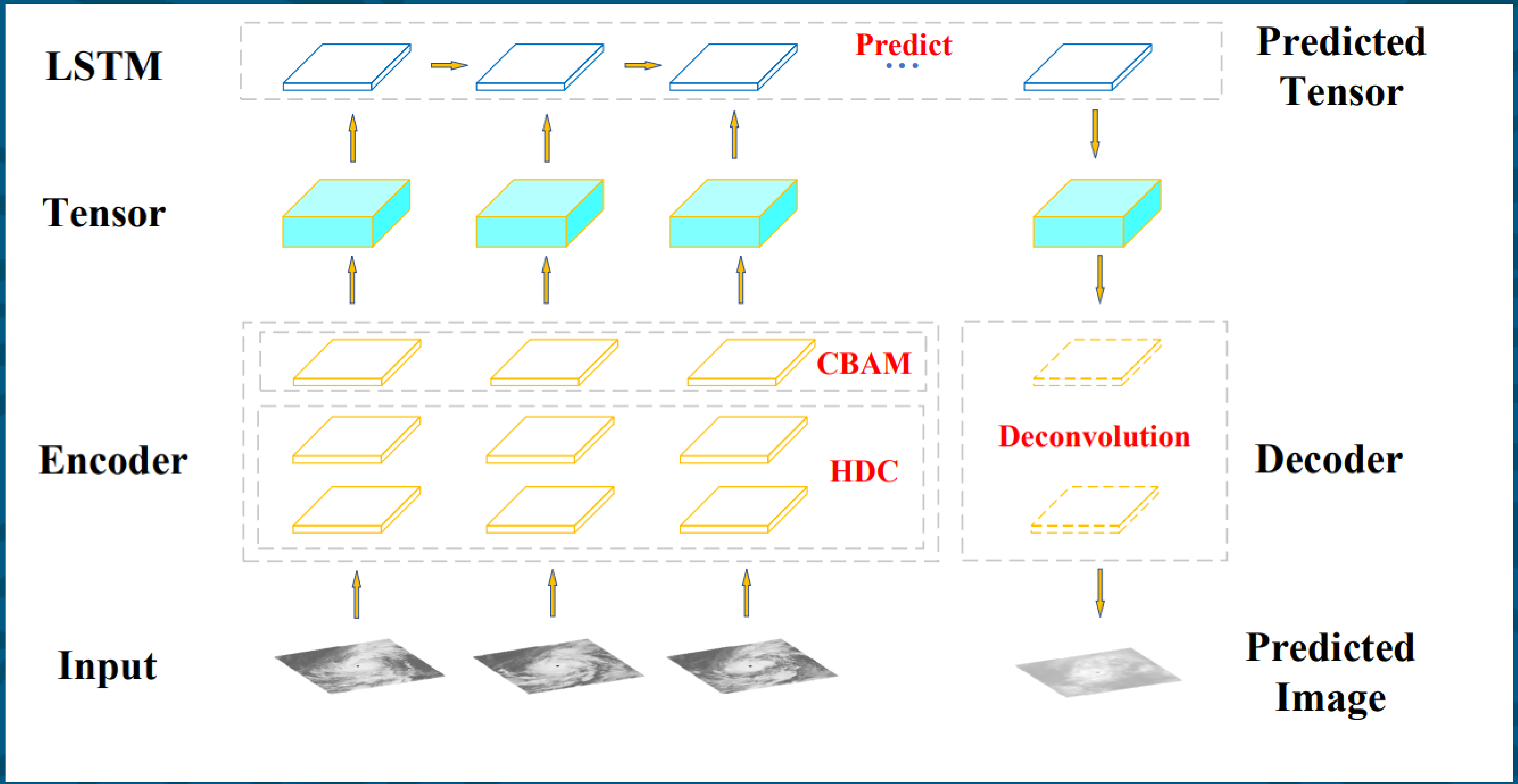
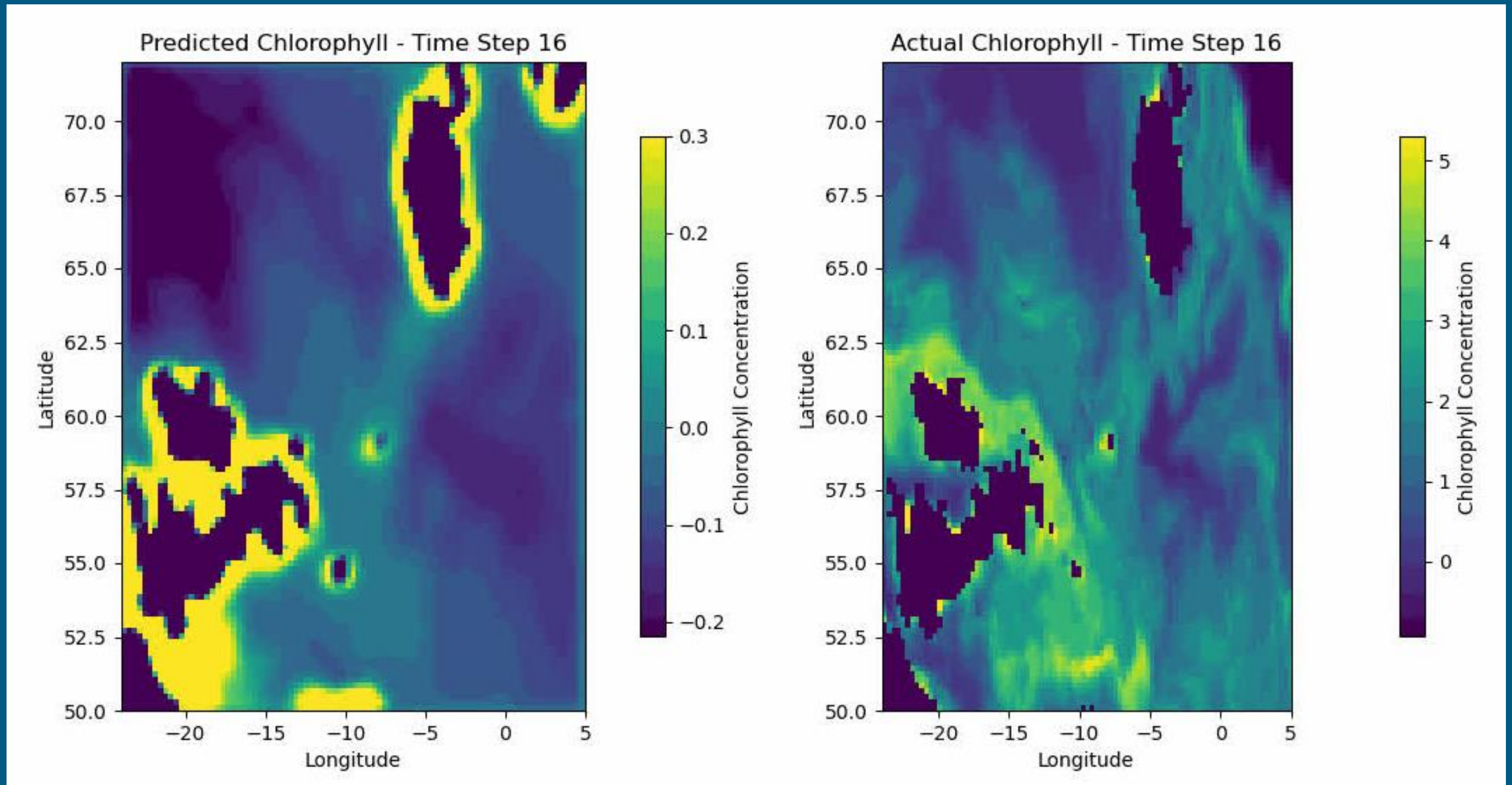
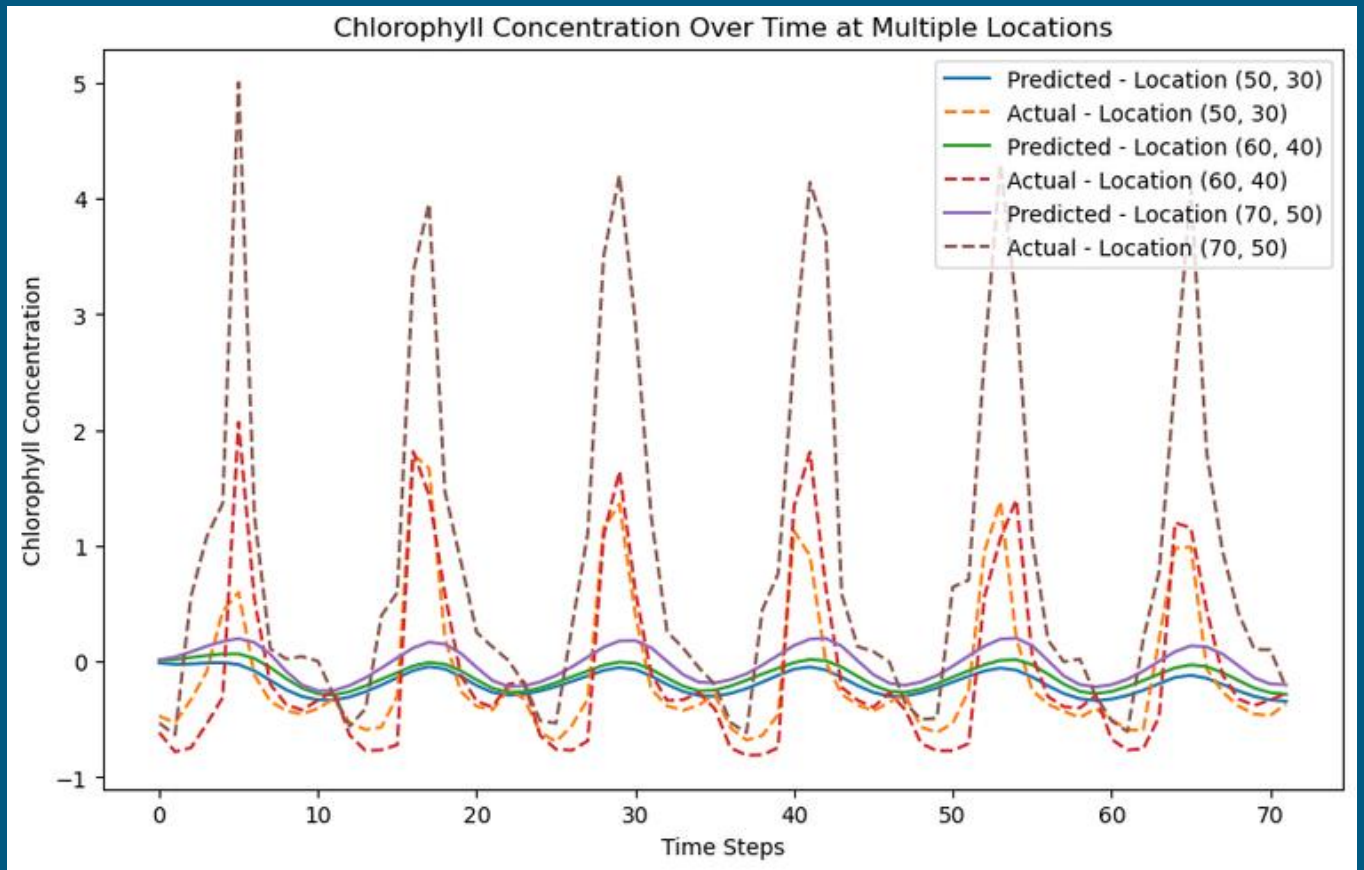


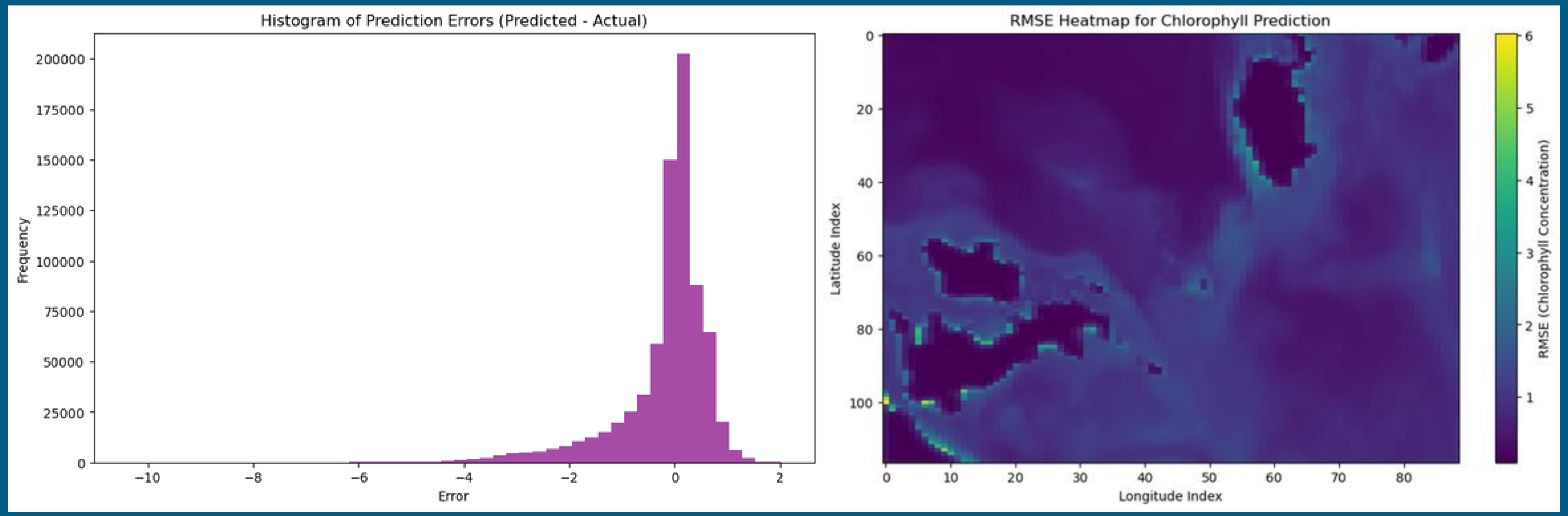
Image and idea found in "A time series image prediction method combining a CNN and LSTM and its application in typhoon track" prediction"

ConvLSTM: Results

- Utilized a ConvLSTM model to capture spatial and temporal dependencies in the chlorophyll concentration data.
- The model was trained for 20 epochs using an Adam optimizer and mixed precision training to speed up the process.
- Evaluated the model using RMSE, which showed some discrepancies between predicted and actual values, particularly in high-concentration areas



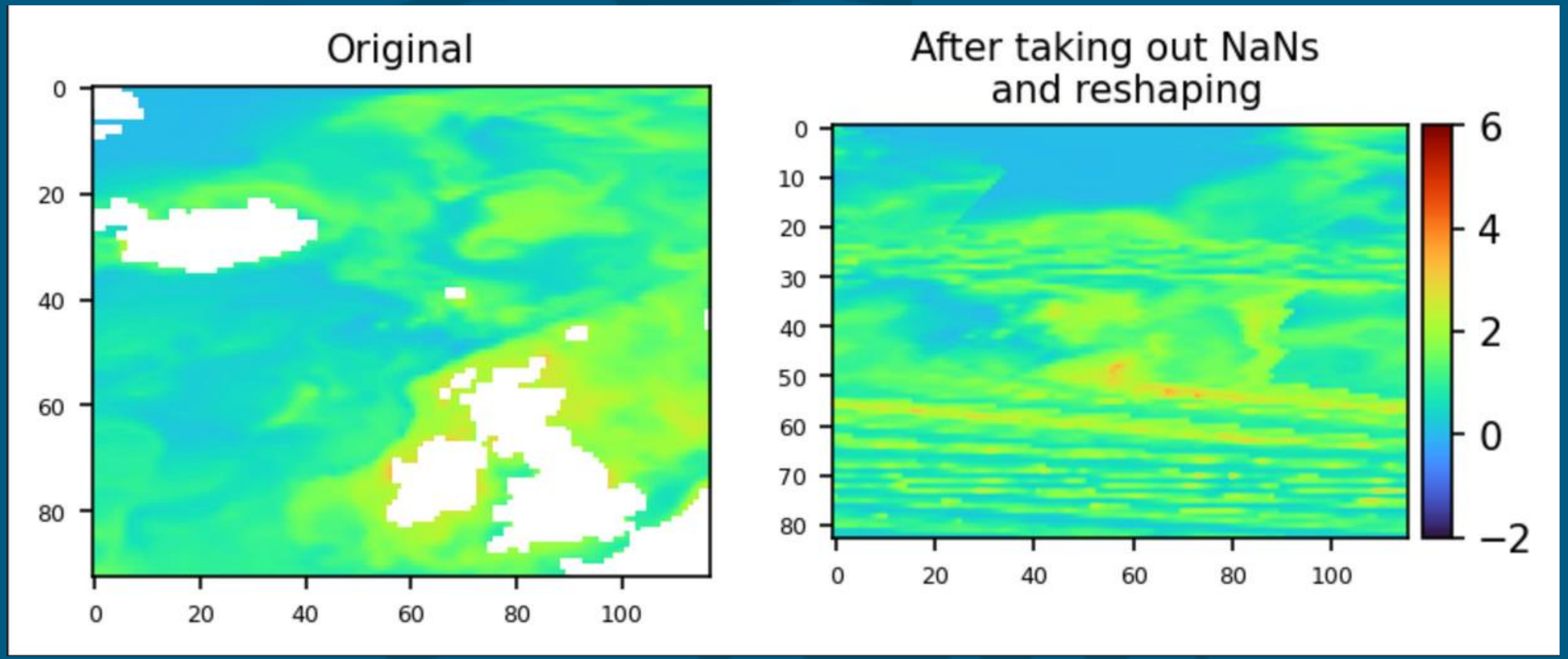




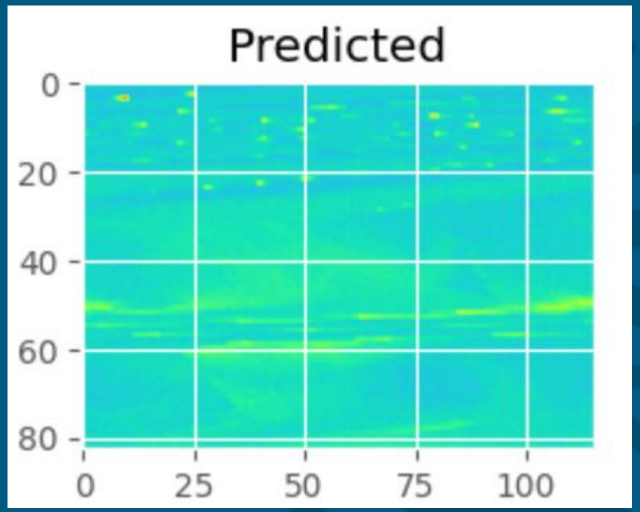
CNN models

- In addition to normalizing training data to have zero mean and unit std, we also process the NaN values as follows:

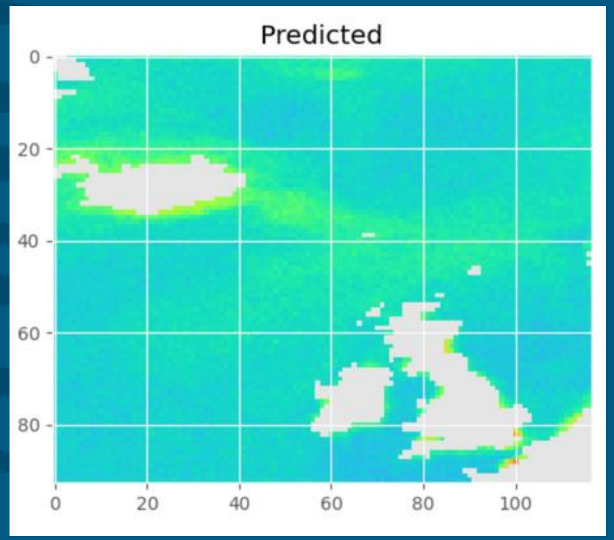
  - Flatten each feature tensor, find and save the positions of NaNs and drop them out.
  - Train the model on the data without NaNs.
  - After prediction plug back the NaNs and get final plot.



Train, validate



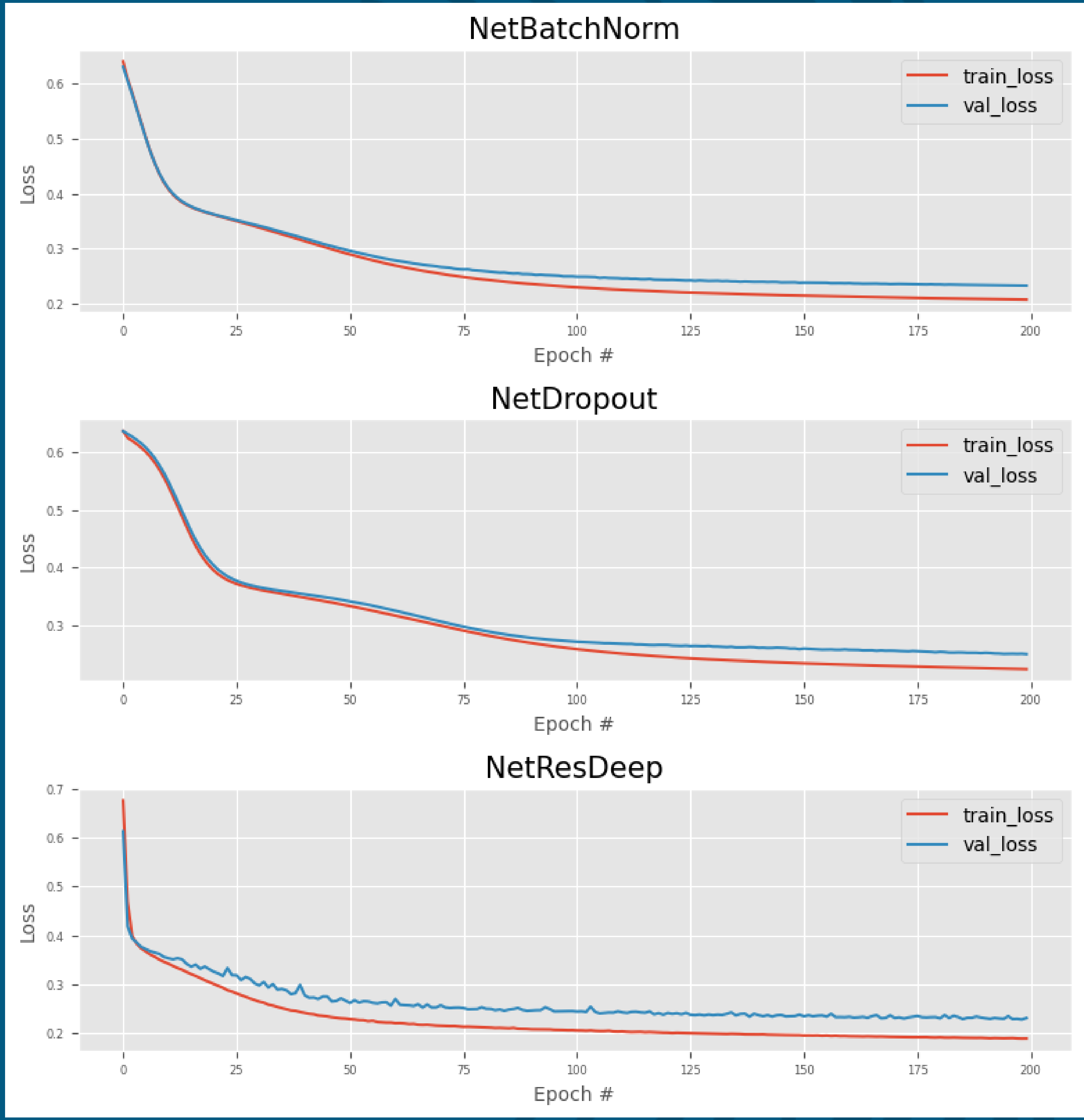
Transform back



Original tensor of 93 x 117 = 10881 is reduced to 83 x 116 = 9628 .

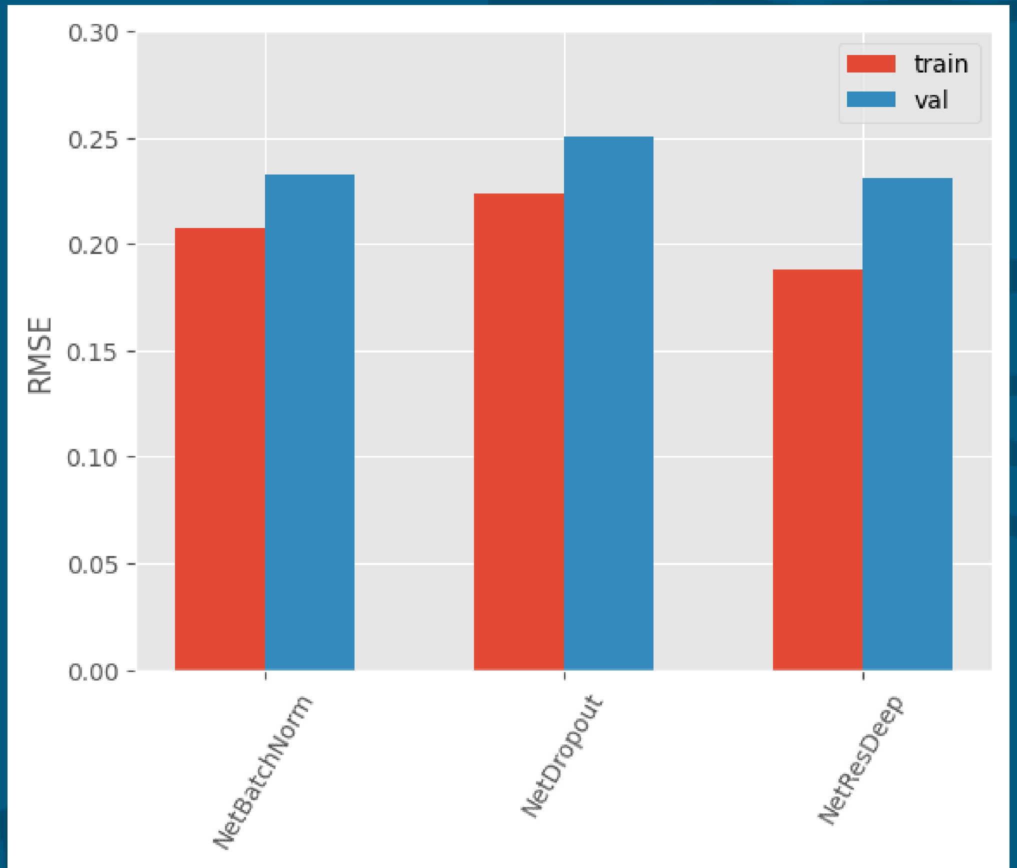
CNN models

Loss/ Epoch plots for each model:



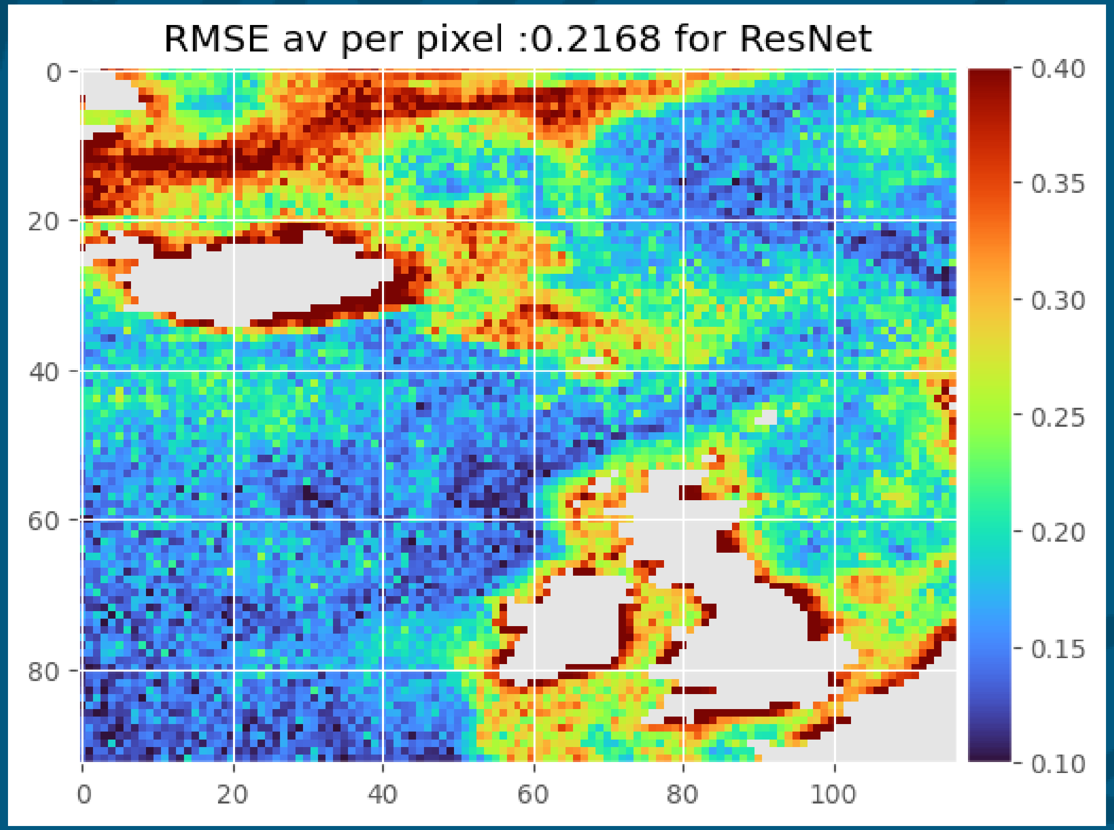

```python
model_resnet = models.NetResDeep(n_chans1=32, ten_size= reduced_tensors.redten_size
                                 , poolten_size=reduced_tensors.poolsize).to(device = device)
optimizer = optim.SGD(params=model_resnet.parameters(), lr=learning_rate)
loss_fn = nn.MSELoss()

model_resnet_list = training.training_loop(
    n_epochs = n_epochs,
    optimizer = optimizer,
    model = model_resnet,
    loss_fn = loss_fn,
    batchsize = batch,
    train_loader = train_loader,
    val_loader = val_loader
```

- We have used ['fe','no3','o2','si','ph'] as features with 288 images (months) for training and 72 images for the validation set.

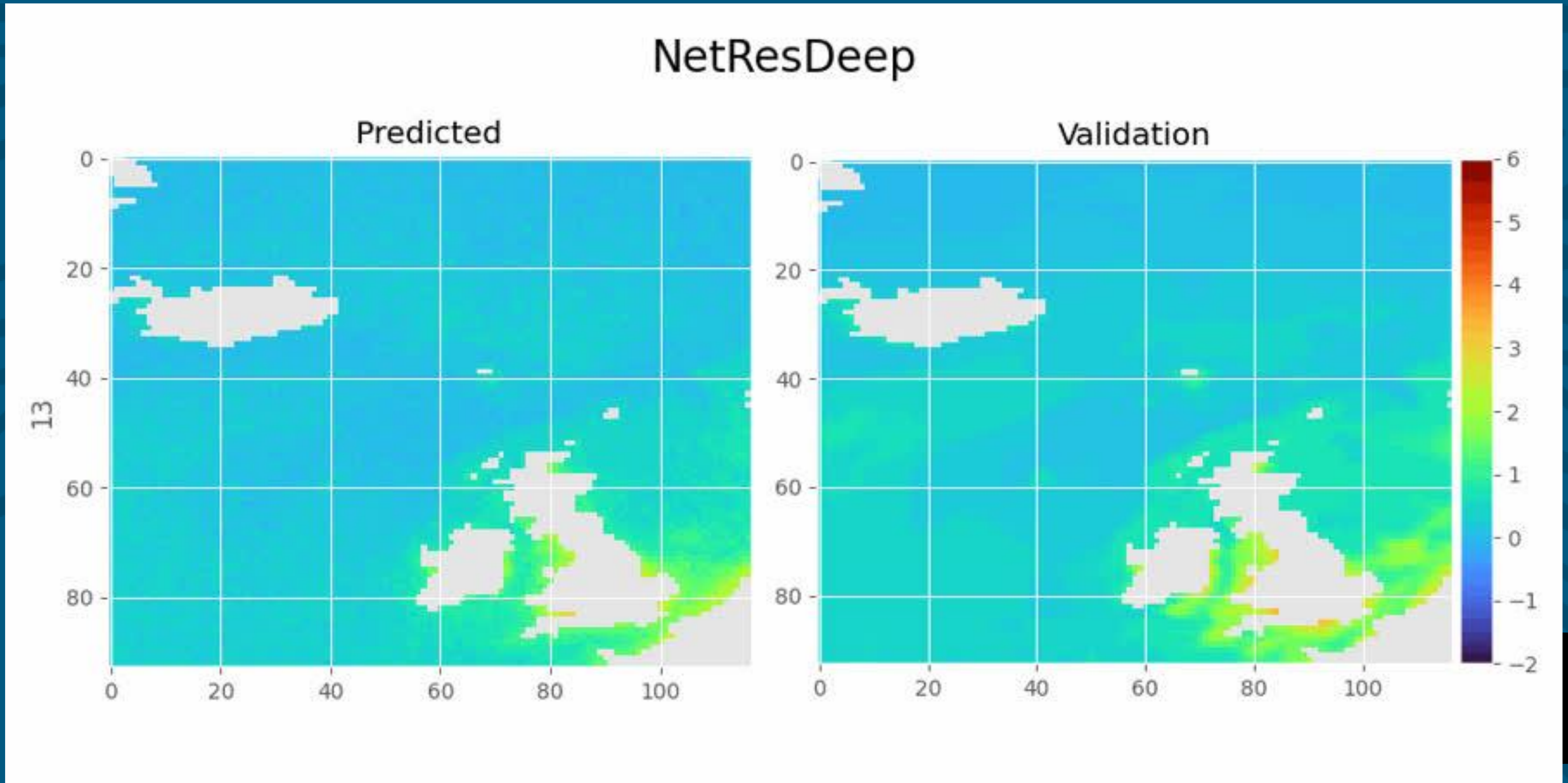


CNN models

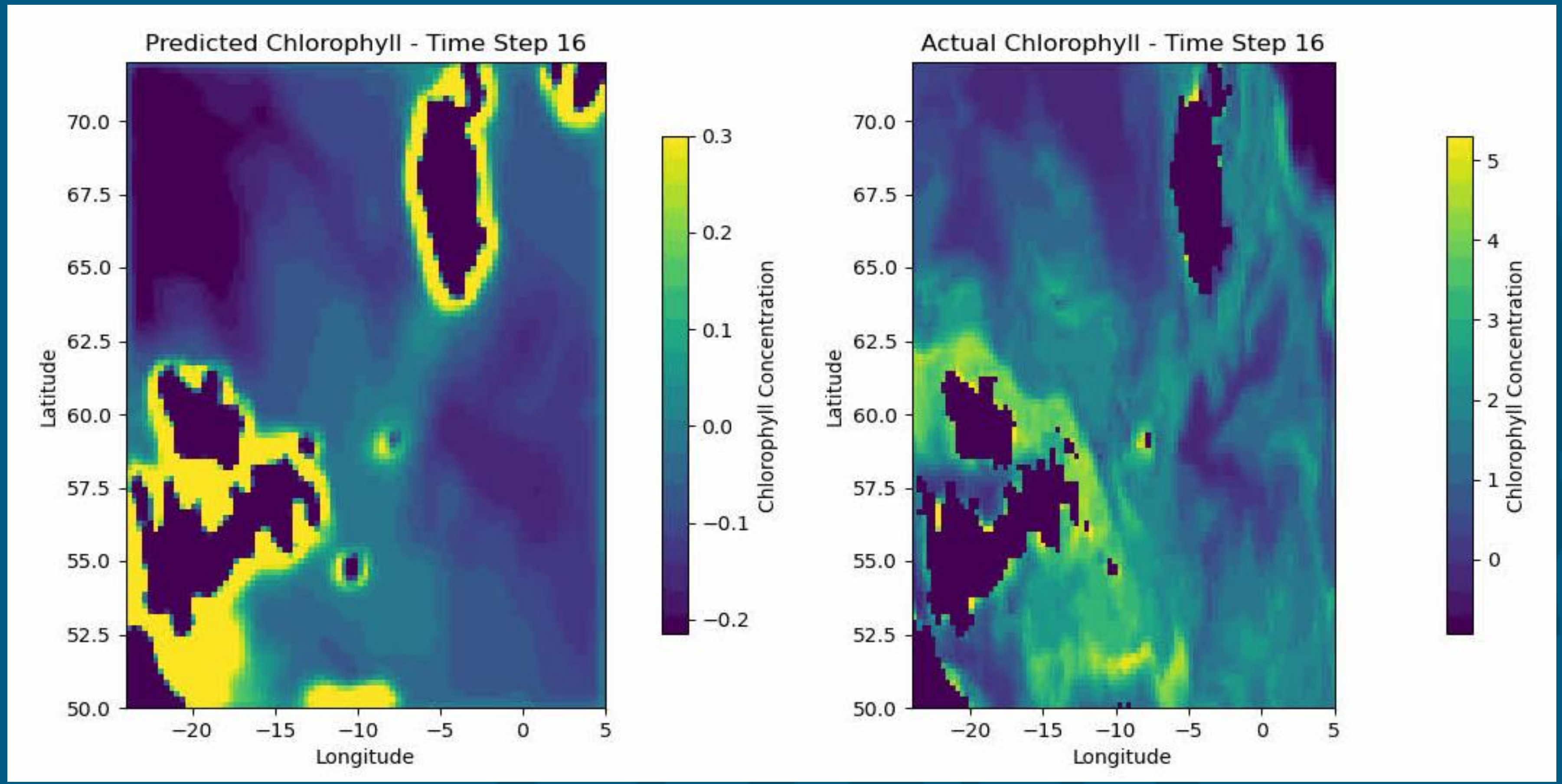
- For each pixel RMSE over 72 months are computed and plotted.

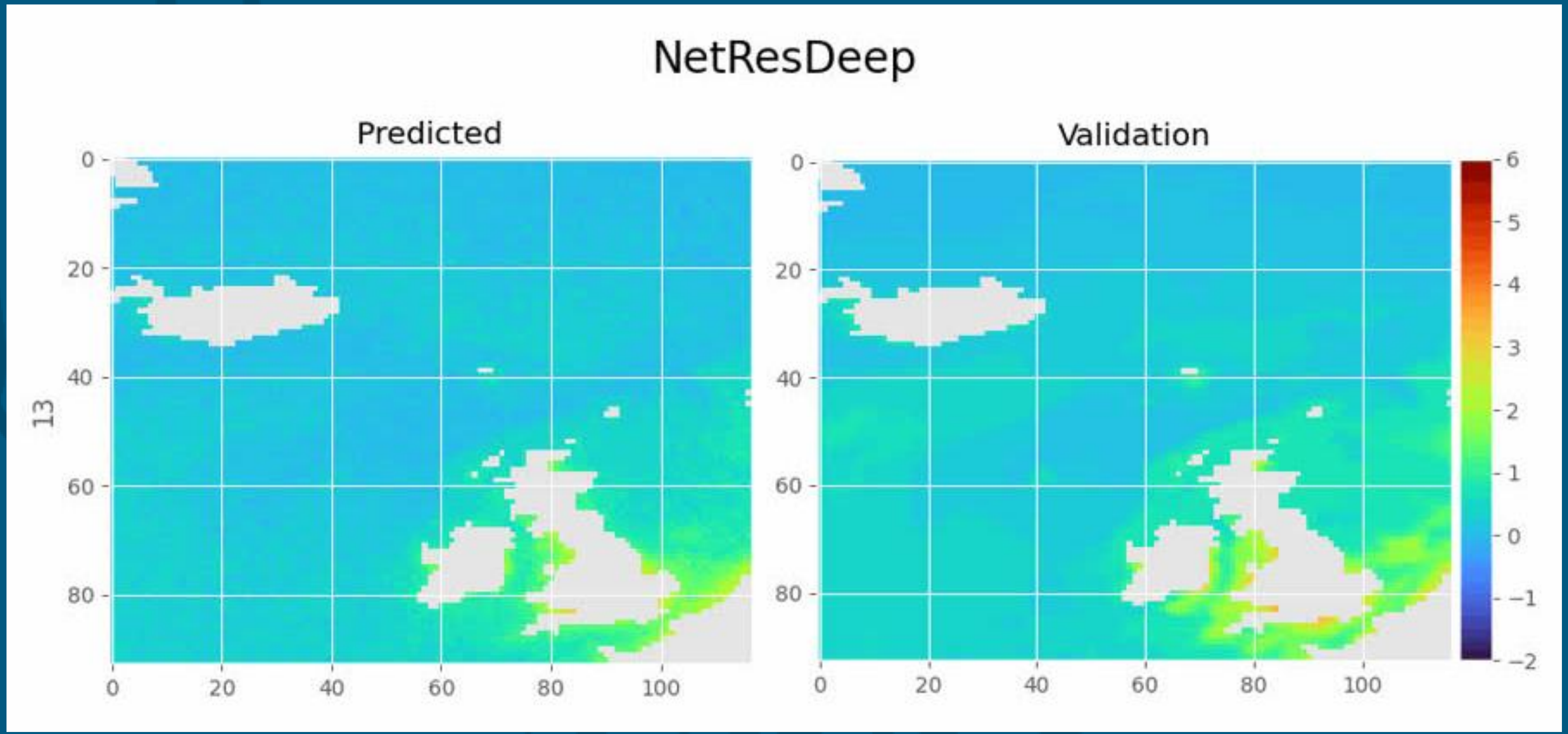


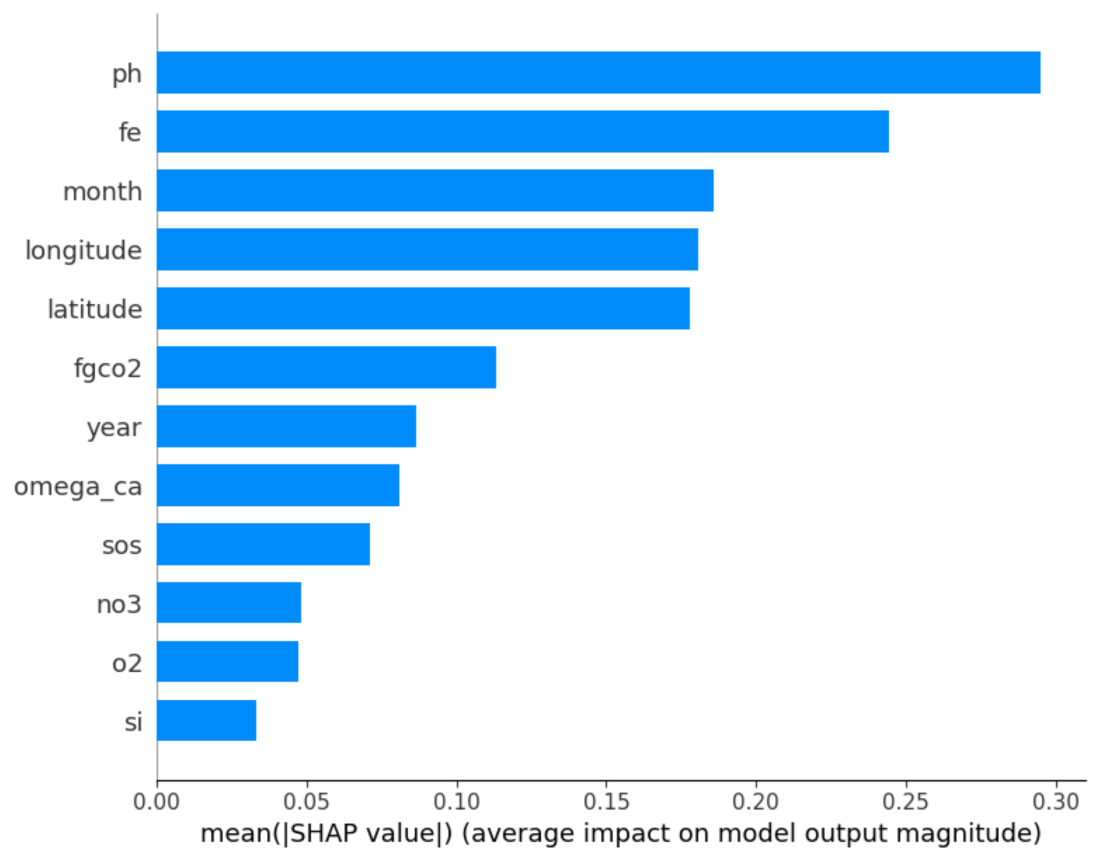
- Predicted results vs the validation set over 72 months.



Conclusions







Model	RMSE
XGBoost	0.32
ResNet	0.22
ConvLSTM	0.89

Iron fertilization can enhance fishery ecosystem

TEAM WUNDERPUS OCTOPUS

NEW ATLANTIS