Performing Finer-Grained Classification of Offshore Marine Platforms

Viraj Goyal
University of Washington
Seattle, WA 98195
vgoyal55@uw.edu

Abstract

Currently, the Satlas project from the Allen Institute for AI is able to identify offshore marine platforms. However, it would be beneficial if the community could understand what these offshore marine platforms actually were. Thus, in this work, we explore finer-grained classification of these offshore marine platforms as a next step to Satlas. We derive finer-grained classes by intersecting U.S. government NOAA ENC data, which includes more specific labels for offshore marine platforms, with current data from Satlas to identify common objects. We then train on these intersected examples, using the label from the NOAA ENC data as our groundtruth finer-grained label for a particular offshore platform Sentinel-2 satellite image that exists in Satlas. Our goal is to achieve 90+% classification accuracy on unseen offshore marine platforms by training a model consisting of an ImageNet-initialized backbone and a randomly initialized feature pyramid network and classification head. We experiment with training on 64x64 images vs. 32x32 images and using a Swin Transformer vs. ResNet50 CNN model backbone. Our experiments demonstrate that the Swin Transformer fine-tuned on 32x32 images performs best, with 86.18% accuracy on the validation set. Despite falling short of our targeted 90+% accuracy, this model derived relevant features for each finer-grained class to perform highly accurate classification of very low-resolution Sentinel-2 imagery, suggesting that our model framework can be adapted to other tasks which only have low-resolution remote sensing data available.

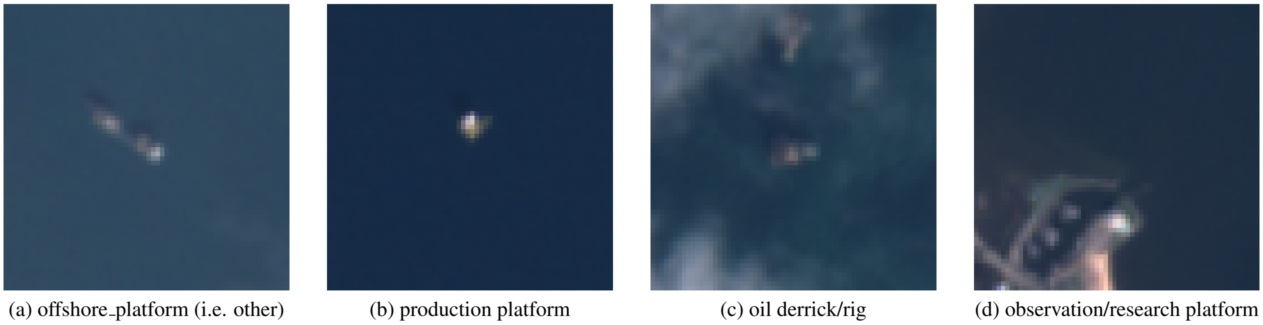
1. Introduction

Satellite imagery offers a wide variety of information about the physical world. Using this imagery and applying novel computer vision techniques, we can map where infrastructure currently is and what type of infrastructure it is. For instance, in images of marine infrastructure such as offshore platforms, we can catalogue where specific platforms are located and for how long, enabling effective asset tracking for safety/emergency responses and even helping earth and environmental scientists garner insights into marine infrastructure. With satellite imagery from the EU's Sentinel missions [1], we can monitor different infrastructure and assets across the Earth.

The Satlas project does exactly that [3]. Currently, Satlas has pre-trained its AI models on the SatlasPretain dataset [2] and then fine-tuned them for classification of marine infrastructure (offshore wind turbines and platforms). Although using low-resolution Sentinel 2 satellite imagery to pinpoint where marine infrastructure is located and classify it into a general category like offshore platforms is impressive, it would be even more powerful if we could perform finer-grained classification of these offshore platforms. Doing so could help us derive where certain resources may be located and/or the intent of these offshore platforms at those locations. For example, recognizing an offshore platform as an oil rig vs. a natural gas rig can help earth scientists understand where these resources might be located and potentially more concentrated in certain places of the Earth.

To derive the potential finer-grained categories within offshore marine platforms, we first observe the NOAA's Electronic Navigational Charts (ENCs) [5], which contain the latitude and longitude of marine objects. We intersect the locations of these objects with the locations of offshore marine platforms detected in Satlas, retaining all the objects from the ENCs with a successful intersection and thus eliminating the objects not visible in Sentinel 2 satellite imagery that Satlas uses. Then, we visually inspect the class IDs of the remaining objects from the ENCs, helping us decide what groups of class IDs can be distinguished from each other and what our finer-grained categories should be.

Next, once we have determined the finer-grained classes and have marked Sentinel 2 satellite training imagery with those finer-grained labels, we fine-tune the Satlas AI model (on GPU machines from AI2 beaker) to classify the low-resolution Sentinel 2 images into these finer-grained categories. After doing so, we compute and analyze the confusion matrix. Since our imagery is low-resolution, our model may struggle classifying images with certain labels, thus



(a) offshore_platform (i.e. other) (b) production platform (c) oil derrick/rig (d) observation/research platform

Figure 1

pushing us to potentially combine certain finer-grained categories and retrain our model.

We iterate on the model by potentially experimenting with different finer-category groupings and model architectures, eventually evaluating each distinct model's performance on a validation set of Sentinel 2 images. The expected outcome is a model with over 90% accuracy in classifying different offshore marine platforms.

2. Related Work

Our research project builds upon what the SatlasPretrain dataset [2] enabled us to do in the field of leveraging remote sensing imagery for classification. Using models pre-trained on the SatlasPretrain dataset, researchers from the Allen Institute for AI were able to teach AI models to better understand geographically and seasonally varied satellite images. One of the tasks they completed with high accuracy was identifying offshore marine platforms and offshore wind turbines. Now, this research project aims to perform even finer-grained classification of those offshore marine platforms. Previous works such as using remote sensing data to perform fine-grained marine ship classification by combining a CNN and Swin Transformer have been done before [4], but our research project specifically focuses on fine-grained offshore platform classification, deriving potential finer-grained categories such as 'oil/derrick rigs', 'production platforms', and 'observation/research platforms'.

3. Methods

3.1. Deriving Usable Finer-grained Categories

After retrieving all the latest raw NOAA ENC data [5] (we used $2/18/2024$ most recently), we wrote a python script to process this data such that there was a .geojson file for each finer-grained category from the NOAA ENC data containing every (latitude, longitude) point coordinate that corresponded to an offshore marine platform classified in that category. The ten finer-grained categories for offshore marine platforms in the NOAA ENC data included 'oil derrick/rig', 'production platform', 'observation/research platform', 'articulated loading platform (ALP)', 'single anchor leg mooring (SALM)', 'mooring tower', 'artificial island', 'floating production, storage and off-loading vessel (FPSO)', 'accommodation platform', and 'navigation, communication and control buoy (NCCB)'. There were some marine objects from the NOAA ENC data that did not have a finer-grained classification attached to them, so we placed them in a catch-all category known as 'offshore_platform.'

Next, we intersected the NOAA ENC data with the most recent offshore platform data from the Satlas project [3] (we used data from January 2024). We labeled a successful intersection as the Satlas project having a corresponding object at the same location as the NOAA ENC data pointed to within 3 decimal places for both latitude and longitude; otherwise, there would be no intersection. We also ensured that each object from Satlas could only match with up to one object from the NOAA ENC data.

After performing this intersection, we realized that the only categories whose NOAA ENC objects intersected $\geq 1\%$ of the time with the objects from Satlas were 'oil derrick/rig', 'production platform', 'accommodation platform', 'mooring tower', 'artificial island', and the catch-all 'offshore_platform' category. Since there were very few (i.e. $1-3$) intersected examples for the 'artificial island', 'mooring tower', and 'accommodation platform' categories, we discarded them.

Now, we were left with the 'oil derrick/rig', 'production platform', and the catch-all 'offshore_platform' category. We realized that training a model on just these three categories, with only two of them being finer-grained, could be tough considering that there were only 22 intersected examples for the 'oil derrick/rig' category.

3.2. Initial Approach

Thus, we remedied our approach. First, we added the category 'observation/research platform' to our filtered categories list above as there were many objects of this category in the NOAA ENC data, even if $< 1\%$ of those ob-

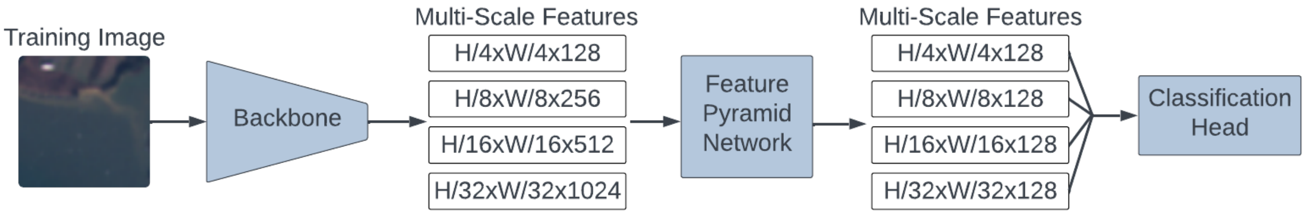


Figure 2. Experiment Models' architecture. After the Classification Head, Softmax is used to derive probabilities for each of the four classes. The input image is predicted as the class with the highest probability score.

Model	Backbone	Training Image Size	Data Augmentations	Top Validation Accuracy
Swin Baseline (32x32)	Swin Transformer	32x32	None	0.8377
Swin (32x32)	Swin Transformer	32x32	Horizontal & Vertical Flip	**0.8618**
Swin (64x64)	Swin Transformer	64x64	Horizontal & Vertical Flip	0.8596
ResNet50 Baseline (32x32)	ResNet50 CNN	32x32	None	0.8004
ResNet50 (32x32)	ResNet50 CNN	32x32	Horizontal & Vertical Flip	0.8092
ResNet50 (64x64)	ResNet50 CNN	64x64	Horizontal & Vertical Flip	0.8399

Table 1. Different models and their corresponding highest accuracies on the validation set of 456 images. Note that in the 'Backbone' column, 'CNN' stands for 'Convolutional Neural Network'.

jects intersected with the objects from Satlas. Then, our initial usable list of finer-grained categories became 'oil derrick/rig', 'production platform', 'observation/research platform', and the catch-all 'offshore_platform' category.

Next, we retrieved all the corresponding Sentinel-2 imagery [1] from January 2024 in size 64x64 with the (latitude, longitude) for all objects under the above categories in the NOAA ENC data centered in each image, even if those objects didn't intersect with any objects from Satlas. However, since we did find many intersections for the 'production platform' and 'offshore_platform' categories, we only employed the imagery corresponding to those intersected objects, but for the other two categories 'oil derrick/rig' and 'observation/research platform', we employed all the imagery when fine-tuning our model since there weren't as many intersections with Satlas. At this point, we had a total of 2279 example Sentinel-2 images to fine-tune a model on. One 64x64 image from each of the above categories is visualized in Figure 1. We split our data into 1824 training images and 456 validation images. Note that the 'offshore_platform' category is known as 'other' when we fine-tuned the model.

3.3. Training Models on Sentinel-2 Imagery

To train our deep learning models using the above remote sensing images, we employed the multisat library, which is internal to the Allen Institute for AI organization.

To achieve our desired 90+% accuracy on the validation set in classifying different offshore marine platforms, we tested two different model backbones and two image sizes to train the model with, keeping the rest of the training configuration the same. The specific experiments are discussed in section 4, and the general training workflow is visualized in Figure 2.

Note that we still denote our overall task of performing finer-grained classification of offshore platforms as a 'fine-tuning' task because we still load our model backbone, the most vital part of our architecture, with the pre-training weights derived by training on ImageNet, although we do initialize the later feature pyramid network intermediate and classification head with random weights. In addition, we do not freeze any layers in our model during training.

Diving deeper into the training configurations, we employed Adam's optimizer, with a learning rate of 0.00001 as it is known to work well empirically within the Satlas project. Our batch size was 16, and we fine-tuned each model for 153 epochs on a NVIDIA RTX A6000 GPU. We also performed data augmentation techniques like Horizontal and Vertical Flip on our training data to help make our model more robust. Our model intermediates included a feature pyramid network, and we employed one classification head total. For our example images, we only provided the RGB channels to our model.

4. Experiments

We fine-tuned six total models to try to achieve over 90% accuracy on the validation set, with two of the models serving as baselines for the two different model backbones we tested. Each of the models was fine-tuned with the configurations specified above, but note that the baseline models for each distinct backbone were fine-tuned on 32x32 images to which no data augmentations were applied so that

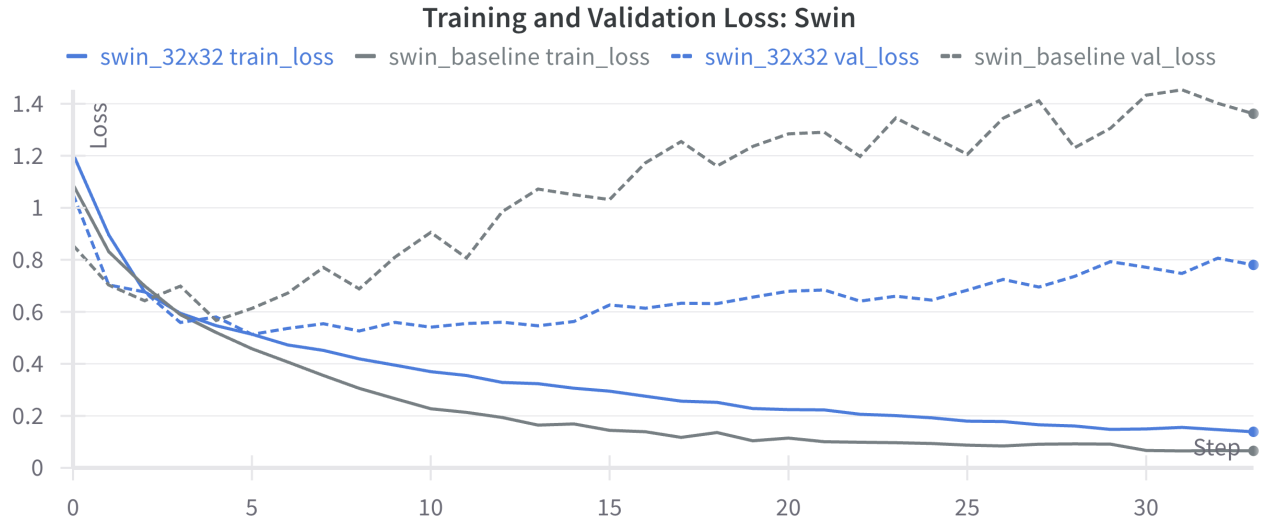


Figure 3. Training and Validation Loss for the best-performing Swin (32x32) model, pictured in blue, and the corresponding Swin Baseline model, pictured in gray.

we could assess the effectiveness of data augmentations as a regularization technique for our scenario.

With regards to the other four models, we fine-tuned two models with a Swin Transformer backbone using 64x64 images and 32x32 images separately and two models with a ResNet50 backbone using 64x64 images and 32x32 images separately. Note that these four models were fine-tuned with the exact configurations specified in 3.3 and that similar methodology in 3.2 can be followed to retrieve the 32x32 images.

Initially, we hypothesized that by fine-tuning a model using 32x32 Sentinel-2 images, with augmentations, centered on the marine object of interest with a Swin Transformer backbone for 153 epochs, we will achieve over 90% accuracy on the validation set and outperform the Swin Baseline model. By using 32x32 Sentinel-2 images, we hoped that the model would be able to learn more enhanced features of the different finer-grained categories because the main objects take up more space in the 32x32 images when compared to the 64x64 images.

Unfortunately, as we can see from Table 1 above (under Figure 2), no model was able to beat the desired 90+% accuracy on the validation set, but we did have some decent initial intuition that the model with the Swin Transformer Backbone fine-tuned on 32x32 images and their augmentations would perform the best on the validation set and outperform the Swin Baseline model. To specify, we obtained a top validation set accuracy of 86.18% for our Swin (32x32) model, beating the corresponding Swin Baseline model's top validation set accuracy by over 2.41%.

5. Discussion

Our findings are promising. Every model crossed 80% validation set accuracy, suggesting that each model was still able to pull distinct-enough features for each finer-grained class, even though the Sentinel-2 images they fine-tuned on were of low resolution. Thus, the methodology in this work could be generalized for other classification tasks involving lower resolution remote sensing data such as illegal vessel detection, forest fire detection, and more. Our work indicates that these models are still able to learn good features for each finer-grained class when fine-tuned on low resolution satellite imagery such that the accuracy on unseen data remains high (i.e. above 80%).

In addition to the above findings, we also observed that every model beat its corresponding baseline model by up to 3.41% when considering top validation accuracy as the comparison metric. On average, our four experimental models beat their corresponding baselines by 2.36%, demonstrating that adding data augmentations was very effective in this scenario. To explain, the horizontal and vertical flip augmentations on the training data exposed our experimental models to a more diverse set of image inputs, which potentially helped the models not only become more robust to variations in the input images but also learn better, more crucial features for each class.

Ultimately, since the Sentinel-2 images are of low resolution, we now realize that surpassing 90+% validation accuracy may be more challenging than we initially thought. Our best model still fell 3.82% short of our 90+% validation accuracy goal. We have listed some causes below.

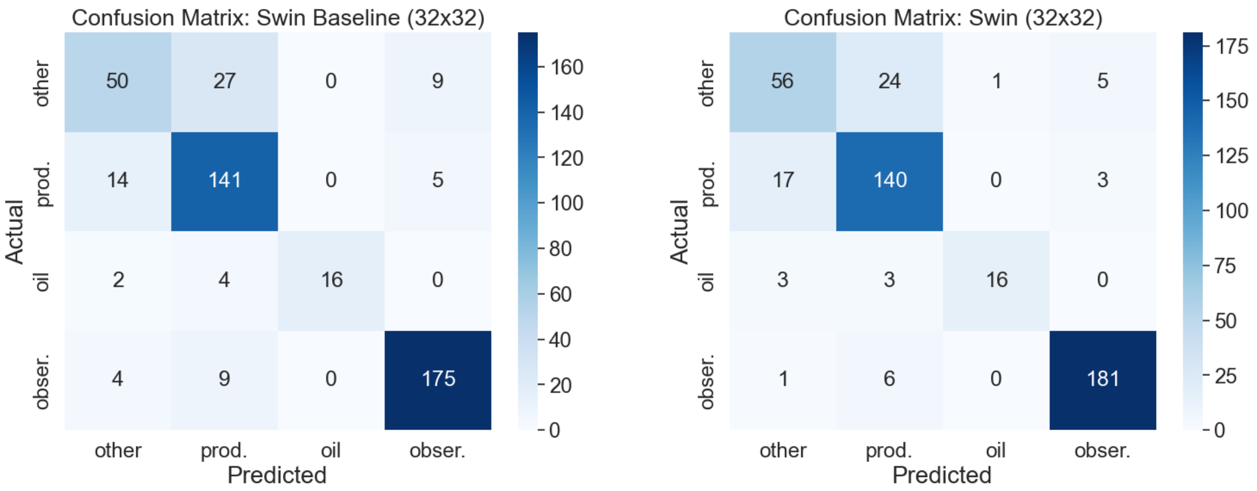


Figure 4. Confusion Matrices on the validation set for the best-performing Swin (32x32) model, pictured second, and the Swin Baseline model, pictured first. Note that these confusion matrices were observed for each model's best performance on the validation set.

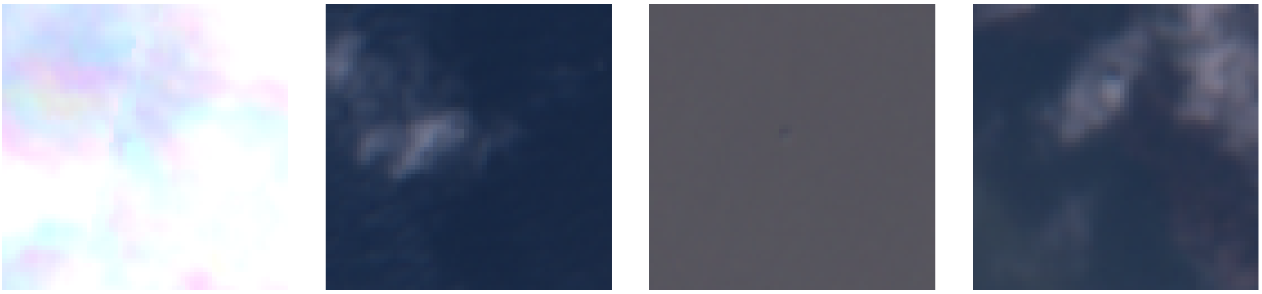


Figure 5. Obscured images in which a platform cannot be easily discerned.

5.1. Potential Overfitting

From Figure 3, we can see that the validation loss for the Swin (32x32) model is at a minimum at step 5, corresponding to after epoch 26 and a validation set accuracy of 81.8%. After this step, training loss continues to decrease, validation loss starts to slowly increase, and top validation set accuracy only improves slightly for the Swin (32x32) model. The above behavior could be a sign of potential overfitting where the model is not generalizing as well to the validation set after step 5 although the model was fine-tuned for over 30 steps. This suggests that we may be training the model for too long in which case the model stops learning more important, underlying features of our classes after epoch 26. To mitigate potential overfitting in the future, we can fine-tune our models for fewer epochs and/or even add further regularization.

5.2. 'Other' Category Ambiguous

From the confusion matrices in Figure 4, we can see that the Swin (32x32) model classified 'observation/research platforms' with a high accuracy of 96.3%, 'oil derrick/rigs' with a low accuracy of 72.7%, 'production platforms' with a medium accuracy of 87.5%, and 'others' with a very low accuracy of 65.1%. Like the best-performing Swin (32x32) model, the Swin Baseline model also encountered lower accuracies with classifying 'oil derrick/rigs' and 'other'. The above observations suggest that some platforms in the 'other' category were not as distinct as previously considered. After taking a further look at the resulting training dataset we created, it appears that some platforms in the 'other' category may have been objects from different, finer-grained classes such as 'production platform' but were never explicitly labeled as so in the original NOAA ENC data. Thus, for examples the model doesn't classify correctly, it may have, in fact, actually been 'correct' in its predictions of the finer-grained class, but since those specific objects were never labeled as that finer-grained class in the training data, its predictions were deemed incorrect, leading to a diminished top validation set accuracy.

To mitigate the ambiguity of the 'other' category, it may be useful to simply remove that category and the associated examples from the training dataset, instead just trying to classify 'production platforms', 'oil derrick/rigs', and 'observation/research platforms' and not including a more redundant catch-all 'other' category. After doing so, we can fine-tune our models on the resulting 'three-class' dataset.

5.3. Obscure Images

From the cloudy and obfuscated images in Figure 5, we can see that there were some examples of images in our dataset in which our models could not easily discern a particular platform. The most common trend in these examples was that the image was too cloudy and/or the platform was too small to recognize in the low-resolution Sentinel-2 imagery. These examples could potentially confuse our models and cause them to learn unwanted features such as 'how much cloud cover there is' to make predictions, but in reality, there was no good signal to begin with for the models to make informed predictions for these examples. One way to mitigate having very obscure and/or cloudy images in the dataset is through manual, human filtering, which is realistic in this scenario considering the entire resulting dataset is only 2279 images. Doing so will decrease the amount of 'poor' data our models are being trained on.

6. Acknowledgements

I would like to express my gratitude to the Allen Institute for AI for mentoring me throughout this work and lending me countless resources like GPUs and internal libraries. Their unwavering support played a pivotal role in bringing this project to fruition. Without them, this work would not have been possible.

References

[1] European Space Agency. Copernicus sentinel missions, 2024. Sentinel 2 Satellite Imagery used to train model. 1, 3

[2] Ritwik Gupta Joe Ferdinando Favyen Bastani, Piper Wolters and Aniruddha Kembhavi. Satlaspretrain: A large-scale dataset for remote sensing image understanding. 2023. 1, 2

[3] The Allen Institute for Artificial Intelligence. satlas.allen.ai, 2024. Satlas Project from Allen Institute for AI. 1, 2

[4] Yalun Zhang Liang Huang, Fengxiang Wang and Qingxia Xu. Fine-grained ship classification by combining cnn and swin transformer. 2022. 2

[5] U.S. Office of Coast Survey. nauticalcharts.noaa.gov, 2024. NOAA ENC data. 1, 2