THERAPY

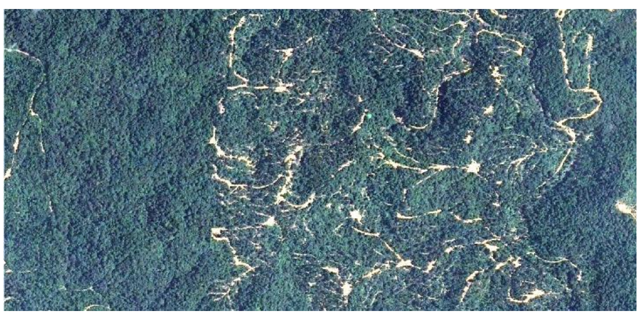
Day 5

The image analytics project prepared here is rather complicated overall. It has been designed specifically based on the use cases of the participants/users. To maximize the assistance that we can provide to all the participants/users, we have prepared a library known as “mysa_lib”. Instead of spending too much time on rewriting or redeveloping the codes, the users can focus on the concept and the values delivered by the library provided.

Image Classifier 1

We start off with a simple classifier we have built for MYSA. This classifier is defined by the author.

Take a look at the following 2 images here:

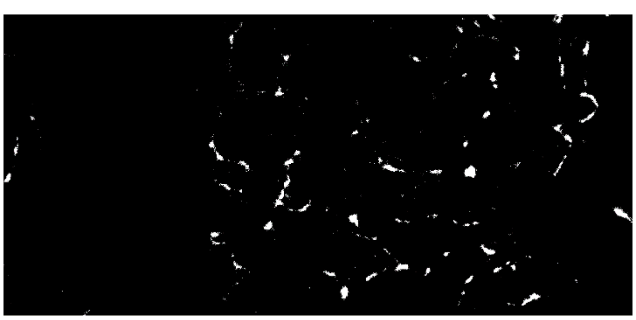


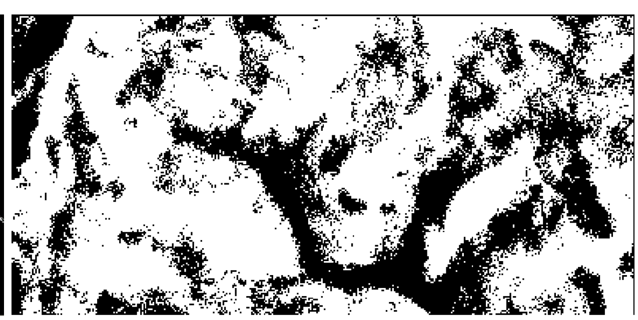
RECOVERINGFOREST.tif LOGGING.tif

So, how do we measure the amount of non-forest regions? The simple_classifier function written specifically for MYSA can handle this easily. Try running the following code.

```
from mysa_lib import simple_classifier
a=simple_classifier (img_file="dataset/RECOVERINGFOREST.tif")
b=simple_classifier (img_file="dataset/LOGGING.tif")
```

```
Percentage of non-forest is 1.2314523544130853
Percentage of non-forest is 66.61416050729028
```

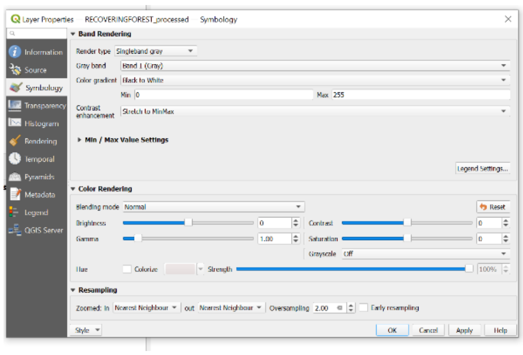


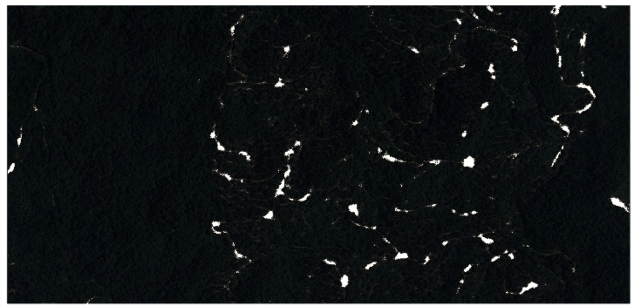


RECOVERINGFOREST_processed.tif LOGGING_processed.tif

Let's look at the result generated. Both "tif" files can be found in the same directory as the input directory. As you can see, the output files are actually mask files generated for the non-forest regions. Also, the percentage of non-forest regions (relative to the given image) has also been given.

For those interested, you can view the result under QGIS.

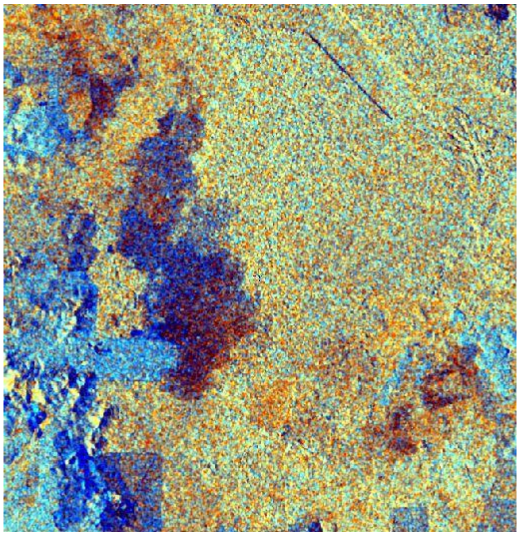
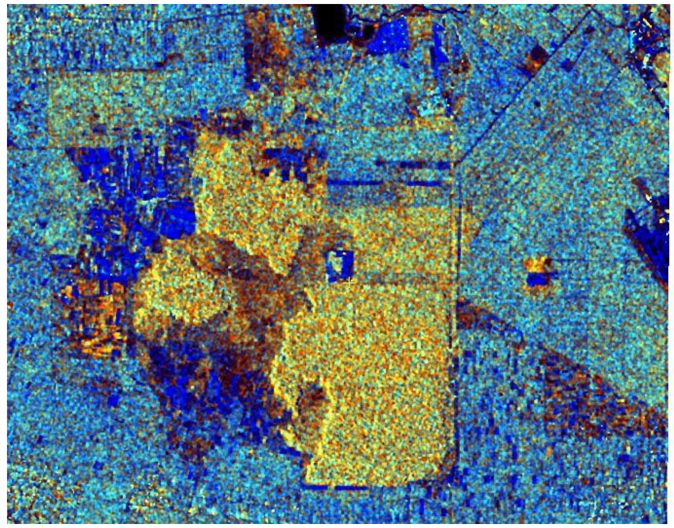




Although the classifier works, but it can only detect the more "severe" non-forest regions. To have more accurate detection, more datasets are required.

Image Classifier 2

Next, we will be looking at a set of Radar images provided by MYSA.

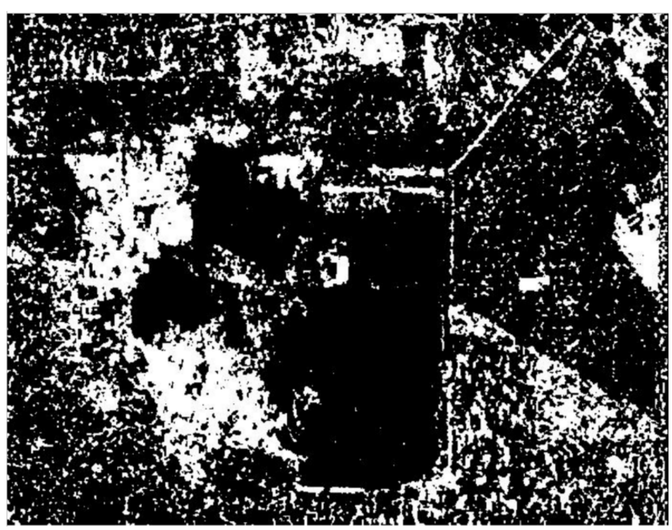



pekan.JPG kuala_langat.JPG

```
from mysa_lib import simple_classifier2

a=simple_classifier2 (img_file="dataset/pekan.JPG")
b=simple_classifier2 (img_file="dataset/kuala_langat.JPG")
```

```
Percentage of blue/dry is 14.999301229101233
Percentage of blue/dry is 25.046680779460456
```

pekan_processed.JPG kuala_langat_processed.JPG

Since we are using JPG images instead of GeoTIFF, there is no geospatial information. In addition, these are only simple classifiers that were developed. Operationalization of these methods require further customization to fit the needs of the users.

AI-based classifier

Next we will look at AI-based detection of non-forest regions. As compared with the previous classifiers, AI-based models/classifiers are known to be more robust. Nonetheless, training of such models requires huge datasets. For the purpose of this training, we will only be using 2 datasets.


20180221_SPOT 6_TasikMuda.tif 20210217_SPOT6_TasikMuda.tif

Observed that the images have some differences in their distribution. For the purpose of this training, the provided images have been normalized (to the best we can).


20180221_SPOT 6_TasikMuda.tif 20210217_SPOT_6_TasikMuda_normalized.tif

For the purpose of this demo, we have already generated the model for you. The model is generated using both the normalized images shown on top. The definition of what is a forest vs. non-forest were extracted from the shapefiles shared by MYSA. With this information, we segment the images into two classes before training the model. Yes, you can now use the model to distinguish the forest from the non-forest class with just a few commands!

For the testing purpose, we will use RECOVERINGFOREST_test.tif, given by MYSA as the test set. This image has not been used in building the model.

```
from mysa_lib import load_image_to_df,model_predict,return_mask,write_mask

pred_image="dataset/RECOVERINGFOREST_test.tif"

raw_df,shape0,shape1=load_image_to_df(raw_file=pred_image)
result=model_predict(raw_df,model_file='dataset/model.json')

classes=["forest","non-forest"]
mask=return_mask(result,shape0,shape1,classes)
write_mask(pred_image,mask)
```

```
Mask file dataset/non_forest_mask.tif is generated
```

Let's look at the result, which is a non_forest_mask.tif, again a mask file (View it under QGIS). The non-forest can be seen to be colored "white". We have used the ground truth as a reference.

As you can see, the model's performance is quite good. It basically never misclassifies non-forest as forest (no false positives). However, it misses out quite a bit on the true non-forest (some degree of false negatives). Performance-wise, it is slightly worse than Image Classifier 1. This result is expected, as the model was built based on two images. If we accumulate a larger dataset, the number of false negatives can be decreased significantly, and the model can perform way better than Image Classifier 1. Of course, this will require manual annotations of the dataset before we can use it to improve the model. In addition, this AI-based classifier can further be improved if there are more bands used instead of just RGB bands. For example, we can incorporate Sentinel satellite images into this model.