Satellite Building Detection & Land Classification

Harold Gamarro

Joe Brian Malubay

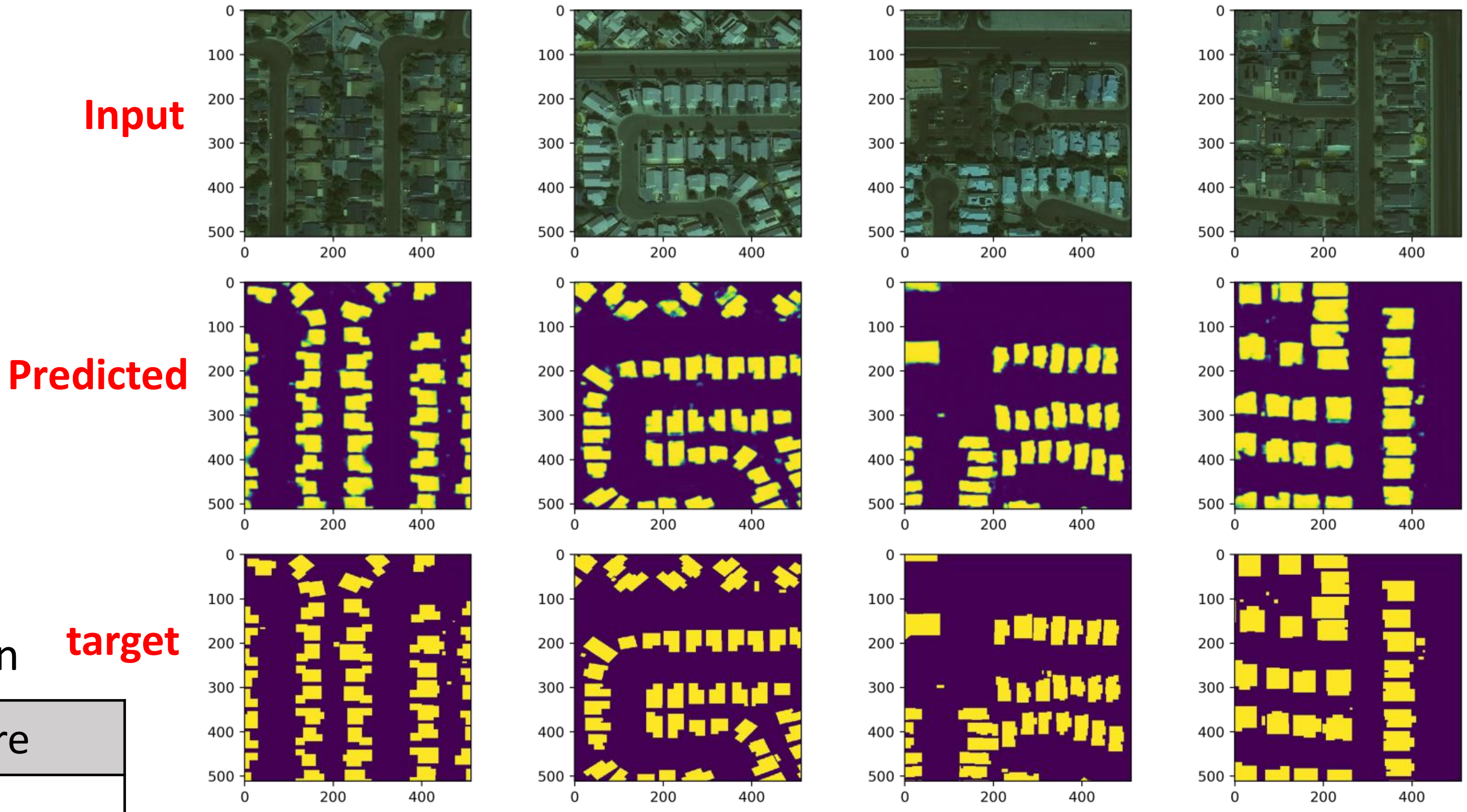
Said Mejia

Juan Pablo Montoya

Building Detection model test #1

- Unet with vgg16 as encoder
- Loss function using
  - Binary Focal Loss
  - Dice Loss
  - Total Loss = dice loss + (1 * focal loss)
- Metrics
  - IOU (Intersection Over Union) Score
  - F1 score
- Input dataset
  - 91 baches of 32x512x512 for training
  - 23 baches of 32x512x512 for validation

Method	mIOU	F1 score
Vgg16 – U-Net	82.84	90.61
Random Forest	In prog.	In prog.
PSPNet	In prog.	In prog.
FPN	In prog.	In prog.
pix2pix	In prog.	In prog.

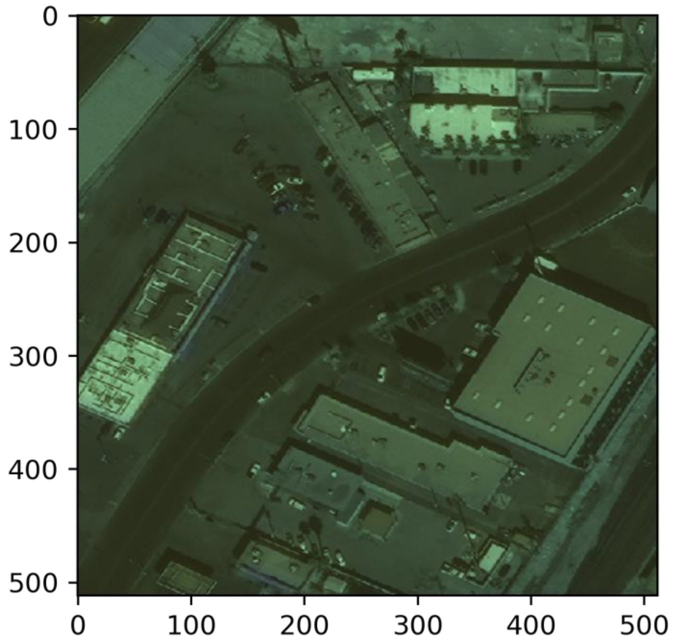


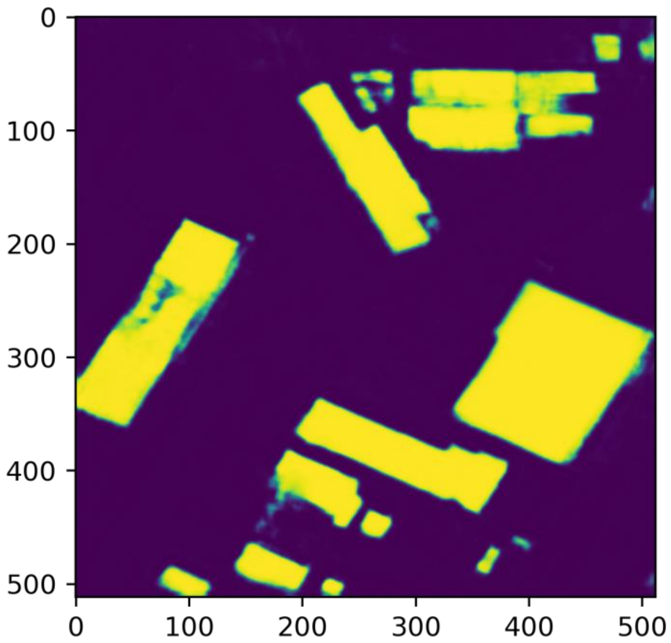
Building Detection model test #1

- Unet with vgg16 as encoder
- Loss function using
  - Binary Focal Loss
  - Dice Loss
  - Total Loss = dice loss + (1 * focal loss)
- Metrics
  - IOU (Intersection Over Union) Score
  - F1 score
- Input dataset
  - 91 baches of 32x512x512 for training
  - 23 baches of 32x512x512 for validation

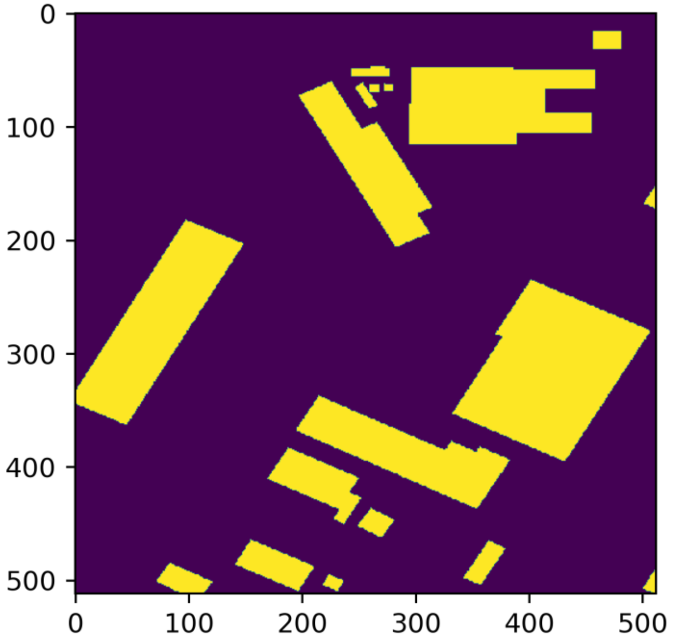
Method	mIOU	F1 score
Vgg16 – U-Net	0.84	0.91
Random Forest	In prog.	In prog.
PSPNet	In prog.	In prog.
FPN	In prog.	In prog.
pix2pix	In prog.	In prog.

Input



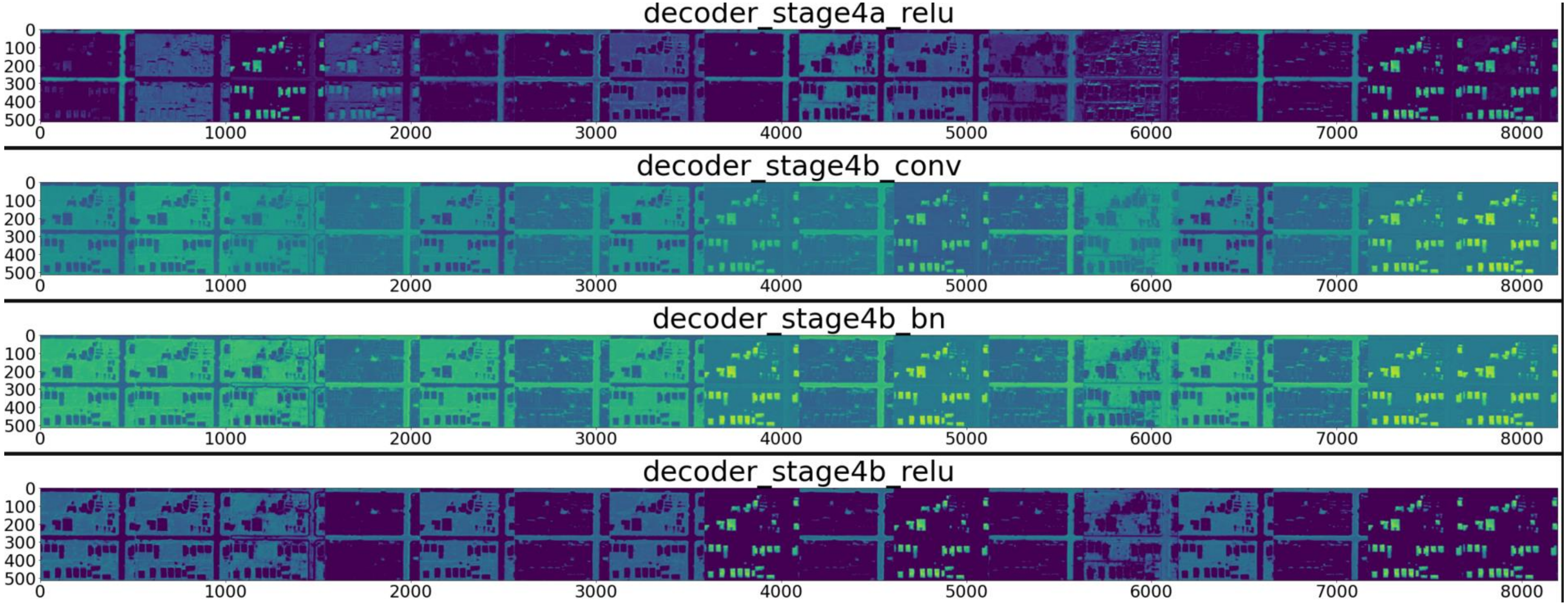


Predicted



target

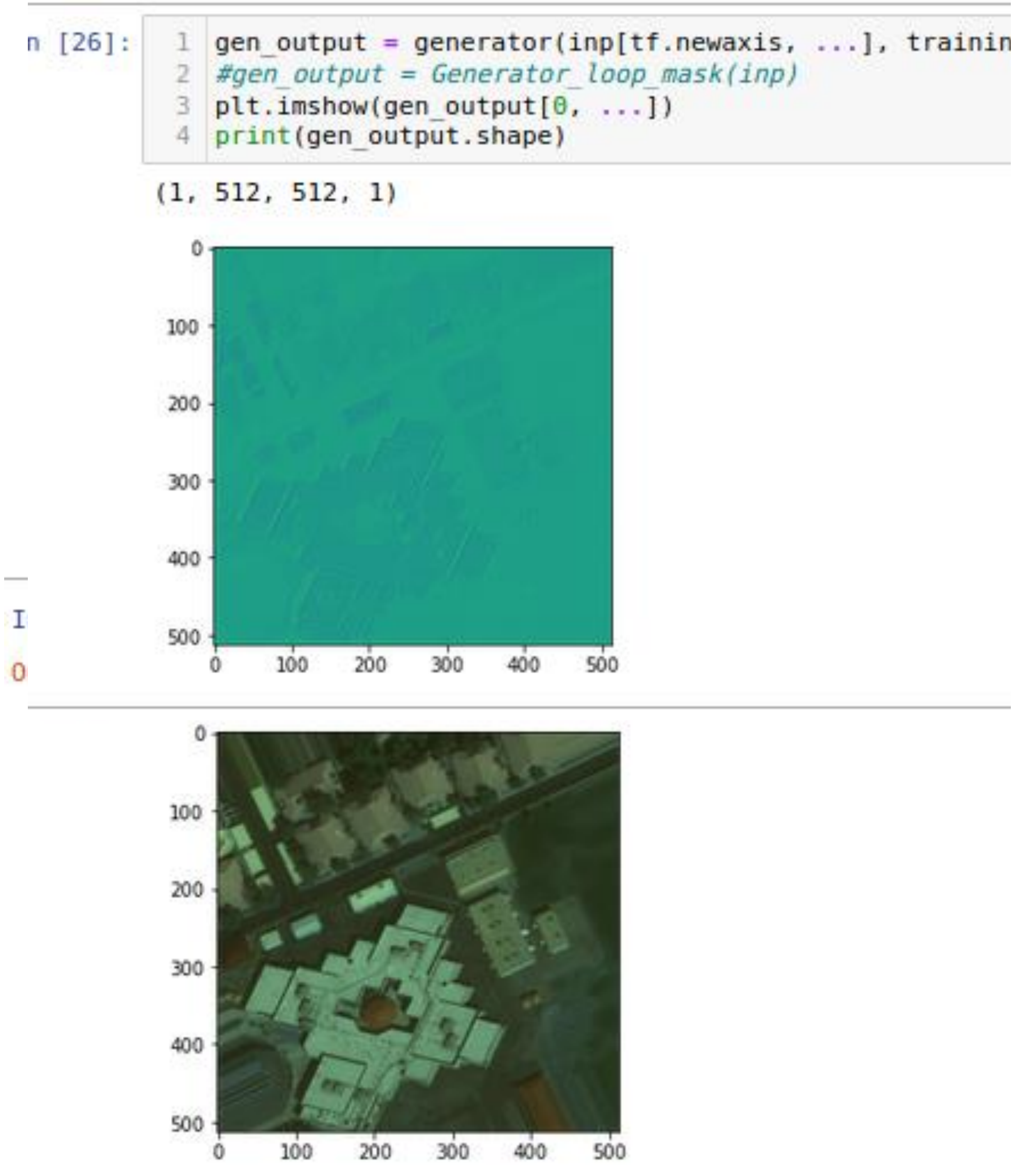
Building Detection model test #1



Intermediate layers example

To do

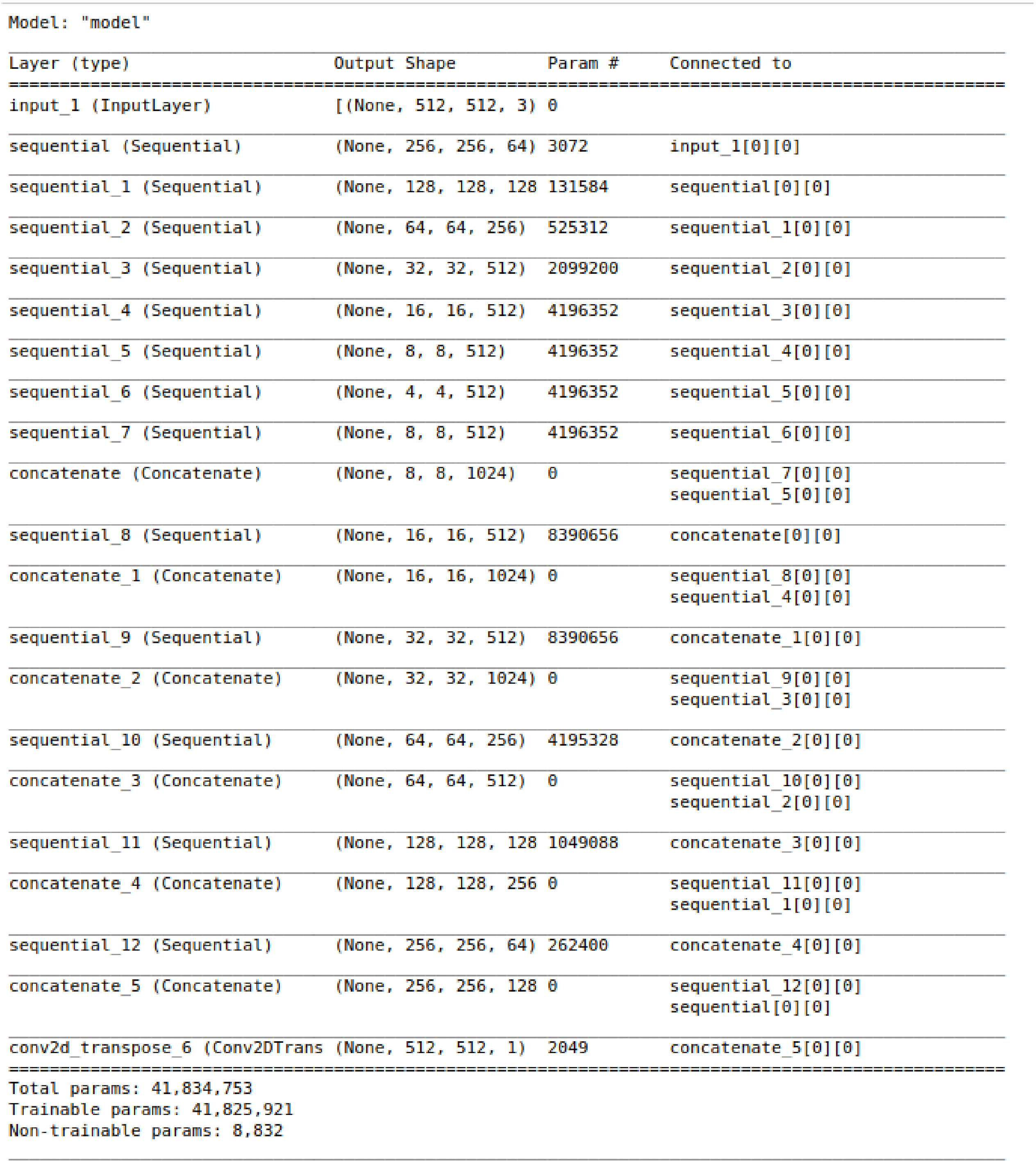
- Run baseline case using random forest
- Setup other CNN models like Pyramid Scene Parsing Network or Feature Pyramid Network
- Setup code to work on GPU cluster because of long train times
- Continue work to visualize intermediate activations



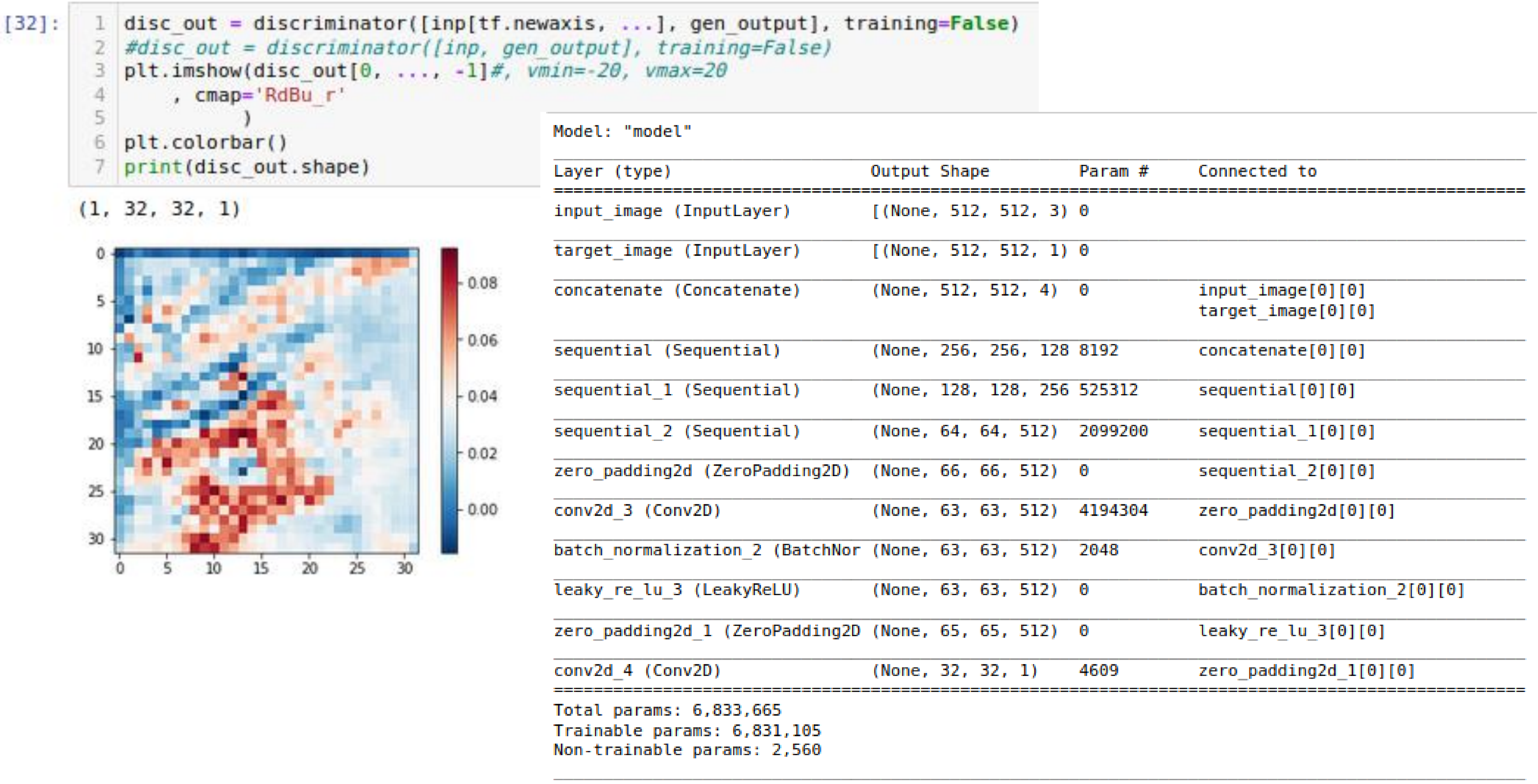
n [26]:

```
gen_output = generator(inp[tf.newaxis, ...], trainin
#gen_output = Generator_loop_mask(inp)
plt.imshow(gen_output[0, ...])
print(gen_output.shape)
```

```
(1, 512, 512, 1)
```



```
Model: "model"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_1 (InputLayer)            [(None, 512, 512, 3) 0
__________________________________________________________________________________________________
sequential (Sequential)         (None, 256, 256, 64) 3072        input_1[0][0]
__________________________________________________________________________________________________
sequential_1 (Sequential)       (None, 128, 128, 128 131584      sequential[0][0]
__________________________________________________________________________________________________
sequential_2 (Sequential)       (None, 64, 64, 256)  525312      sequential_1[0][0]
__________________________________________________________________________________________________
sequential_3 (Sequential)       (None, 32, 32, 512)  2099200     sequential_2[0][0]
__________________________________________________________________________________________________
sequential_4 (Sequential)       (None, 16, 16, 512)  4196352     sequential_3[0][0]
__________________________________________________________________________________________________
sequential_5 (Sequential)       (None, 8, 8, 512)    4196352     sequential_4[0][0]
__________________________________________________________________________________________________
sequential_6 (Sequential)       (None, 4, 4, 512)    4196352     sequential_5[0][0]
__________________________________________________________________________________________________
sequential_7 (Sequential)       (None, 8, 8, 512)    4196352     sequential_6[0][0]
__________________________________________________________________________________________________
concatenate (Concatenate)       (None, 8, 8, 1024)   0           sequential_7[0][0]
                                                                 sequential_5[0][0]
__________________________________________________________________________________________________
sequential_8 (Sequential)       (None, 16, 16, 512)  8390656     concatenate[0][0]
__________________________________________________________________________________________________
concatenate_1 (Concatenate)     (None, 16, 16, 1024) 0           sequential_8[0][0]
                                                                 sequential_4[0][0]
__________________________________________________________________________________________________
sequential_9 (Sequential)       (None, 32, 32, 512)  8390656     concatenate_1[0][0]
__________________________________________________________________________________________________
concatenate_2 (Concatenate)     (None, 32, 32, 1024) 0           sequential_9[0][0]
                                                                 sequential_3[0][0]
__________________________________________________________________________________________________
sequential_10 (Sequential)      (None, 64, 64, 256)  4195328     concatenate_2[0][0]
__________________________________________________________________________________________________
concatenate_3 (Concatenate)     (None, 64, 64, 512)  0           sequential_10[0][0]
                                                                 sequential_2[0][0]
__________________________________________________________________________________________________
sequential_11 (Sequential)      (None, 128, 128, 128 1049088     concatenate_3[0][0]
__________________________________________________________________________________________________
concatenate_4 (Concatenate)     (None, 128, 128, 256 0           sequential_11[0][0]
                                                                 sequential_1[0][0]
__________________________________________________________________________________________________
sequential_12 (Sequential)      (None, 256, 256, 64) 262400      concatenate_4[0][0]
__________________________________________________________________________________________________
concatenate_5 (Concatenate)     (None, 256, 256, 128 0           sequential_12[0][0]
                                                                 sequential[0][0]
__________________________________________________________________________________________________
conv2d_transpose_6 (Conv2DTrans (None, 512, 512, 1)  2049        concatenate_5[0][0]
==================================================================================================
Total params: 41,834,753
Trainable params: 41,825,921
Non-trainable params: 8,832
__________________________________________________________________________________________________
```

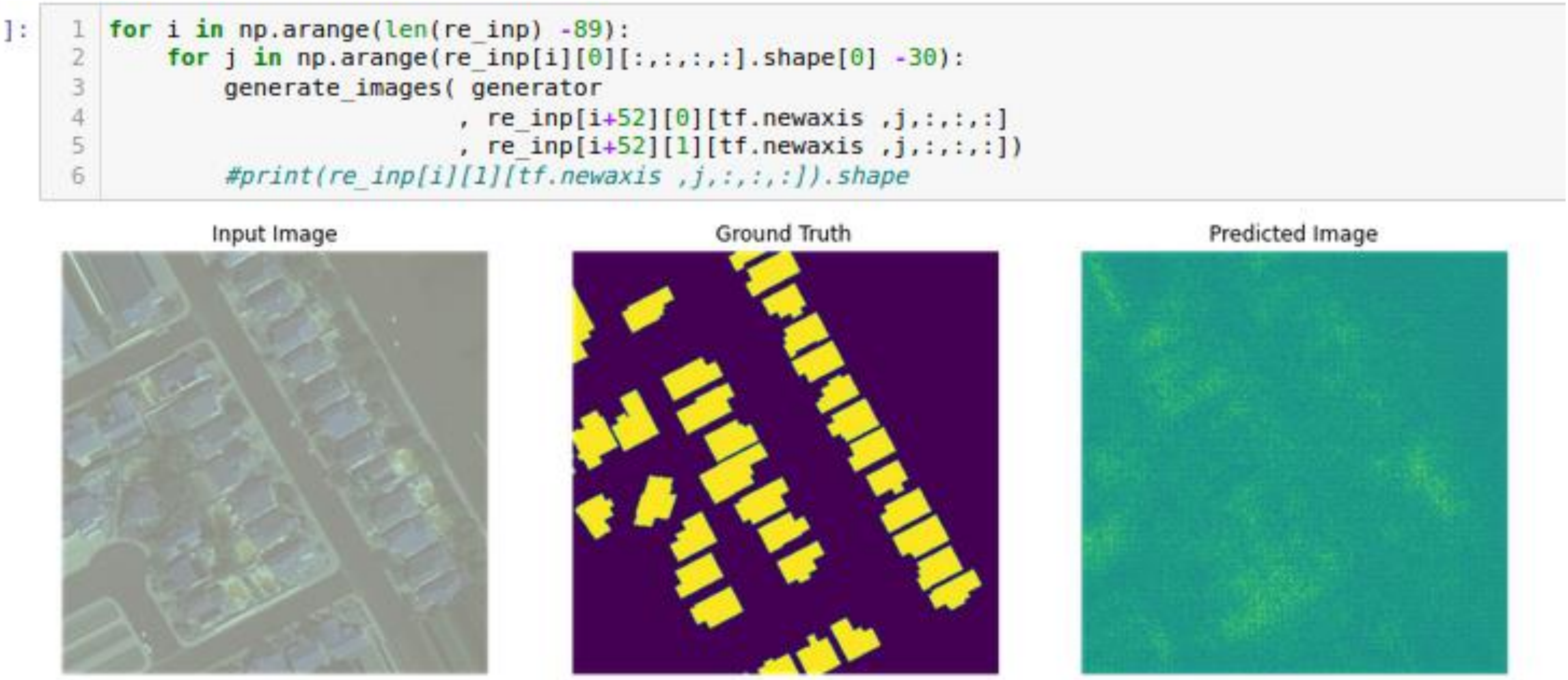


[32]:

```
disc_out = discriminator([inp[tf.newaxis, ...], gen_output], training=False)
#disc_out = discriminator([inp, gen_output], training=False)
plt.imshow(disc_out[0, ..., -1]#, vmin=-20, vmax=20
    , cmap='RdBu_r'
          )
plt.colorbar()
print(disc_out.shape)
```

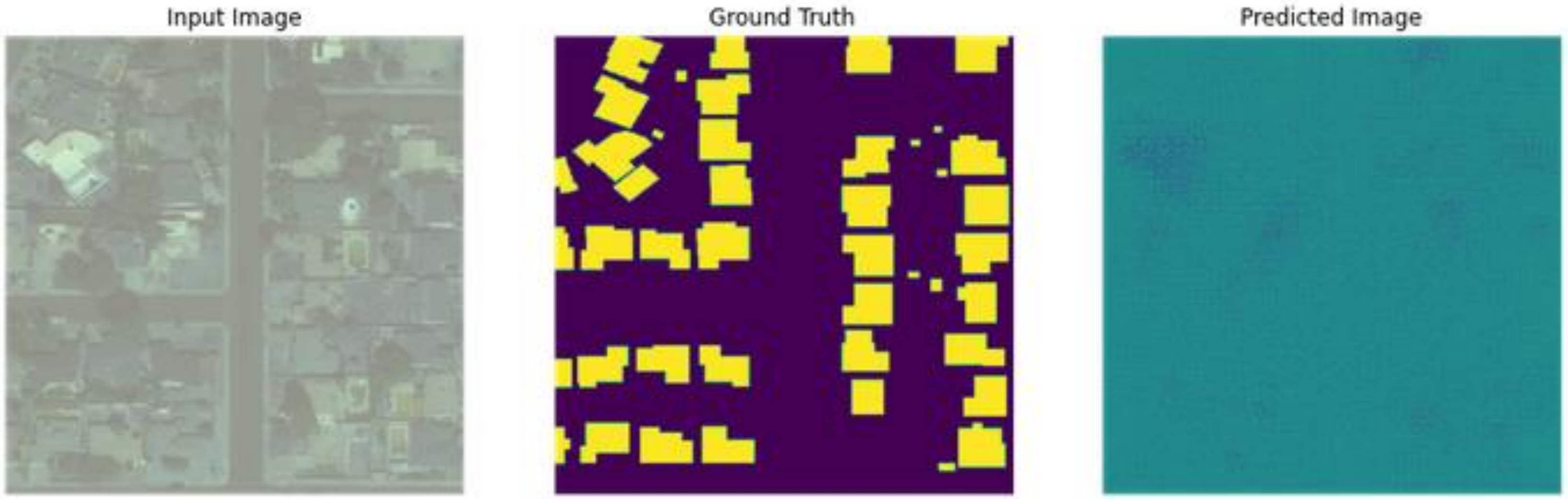
```
(1, 32, 32, 1)
```

```
Model: "model"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_image (InputLayer)        [(None, 512, 512, 3) 0
__________________________________________________________________________________________________
target_image (InputLayer)       [(None, 512, 512, 1) 0
__________________________________________________________________________________________________
concatenate (Concatenate)       (None, 512, 512, 4)  0           input_image[0][0]
                                                                 target_image[0][0]
__________________________________________________________________________________________________
sequential (Sequential)         (None, 256, 256, 128 8192        concatenate[0][0]
__________________________________________________________________________________________________
sequential_1 (Sequential)       (None, 128, 128, 256 525312      sequential[0][0]
__________________________________________________________________________________________________
sequential_2 (Sequential)       (None, 64, 64, 512)  2099200     sequential_1[0][0]
__________________________________________________________________________________________________
zero_padding2d (ZeroPadding2D)  (None, 66, 66, 512)  0           sequential_2[0][0]
__________________________________________________________________________________________________
conv2d_3 (Conv2D)               (None, 63, 63, 512)  4194304     zero_padding2d[0][0]
__________________________________________________________________________________________________
batch_normalization_2 (BatchNor (None, 63, 63, 512)  2048        conv2d_3[0][0]
__________________________________________________________________________________________________
leaky_re_lu_3 (LeakyReLU)       (None, 63, 63, 512)  0           batch_normalization_2[0][0]
__________________________________________________________________________________________________
zero_padding2d_1 (ZeroPadding2D (None, 65, 65, 512)  0           leaky_re_lu_3[0][0]
__________________________________________________________________________________________________
conv2d_4 (Conv2D)               (None, 32, 32, 1)    4609        zero_padding2d_1[0][0]
==================================================================================================
Total params: 6,833,665
Trainable params: 6,831,105
Non-trainable params: 2,560
__________________________________________________________________________________________________
```

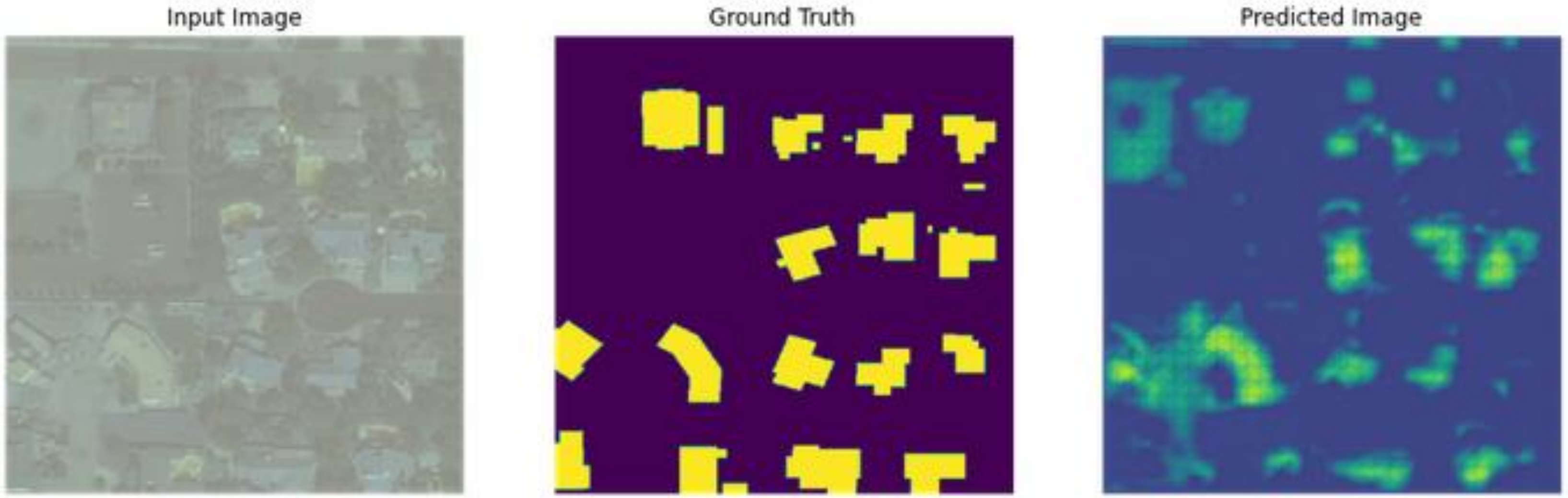


```
for i in np.arange(len(re_inp) -89):
    for j in np.arange(re_inp[i][0][:,:,:,:].shape[0] -30):
        generate_images( generator
                        , re_inp[i+52][0][tf.newaxis ,j,:,:,:]
                        , re_inp[i+52][1][tf.newaxis ,j,:,:,:])
        #print(re_inp[i][1][tf.newaxis ,j,:,:,:]).shape
```

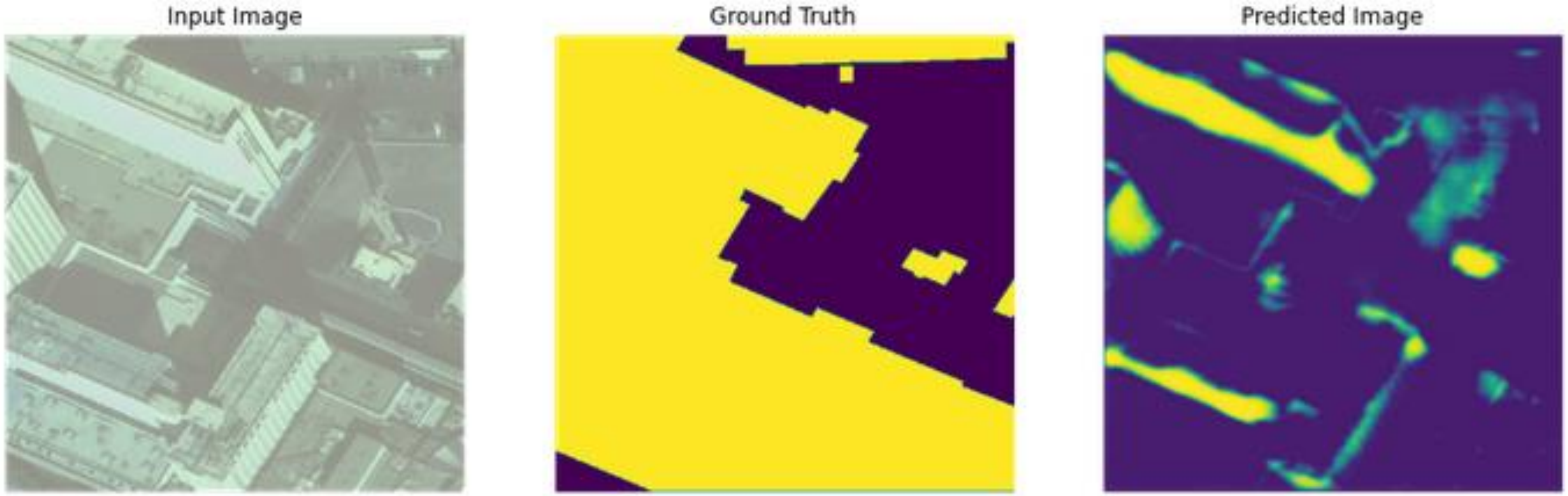
for example_input, example_target in test_dataset.take(1): generate_images(generator, example_input, example_target)



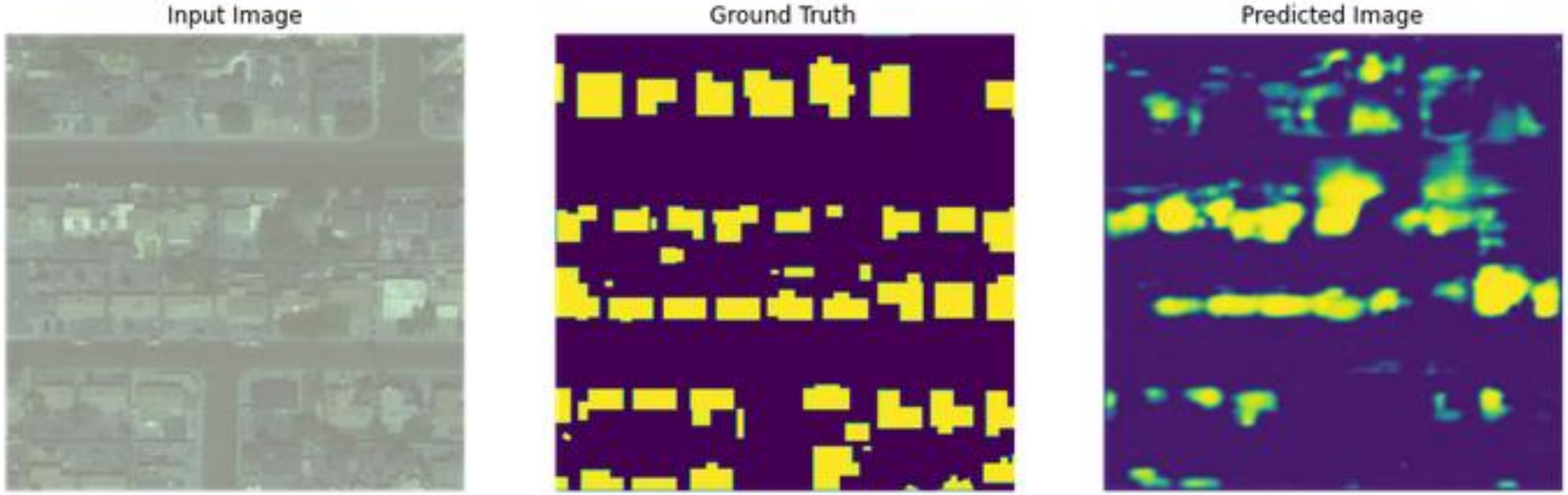
```
.(1, 512, 512, 1)
(1, 512, 512, 1)
................................-finished a training batch-

--------starting next validation image-
```



```
.................................-finished a training batch-
```

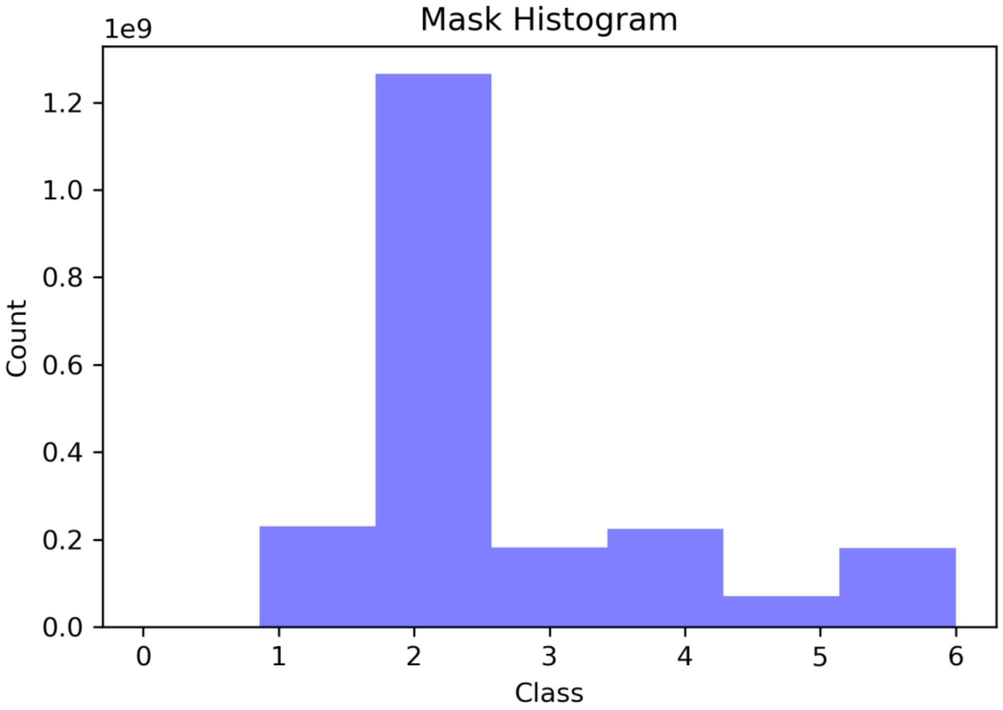


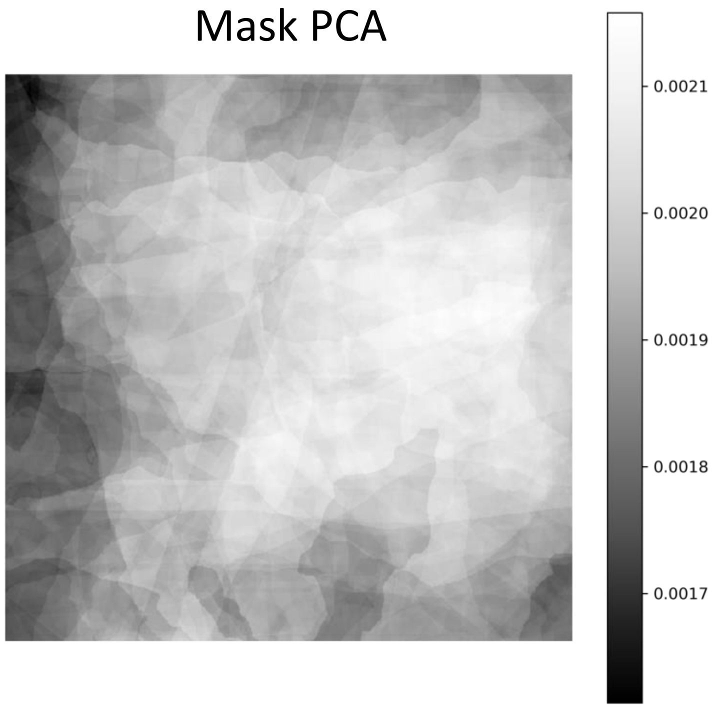
```
.................................-finished a training batch-

--------starting next validation image-

-----------------------finished a validation batch-
```



```
.................................-finished a training batch-
```

Land Classification histogram and PCA

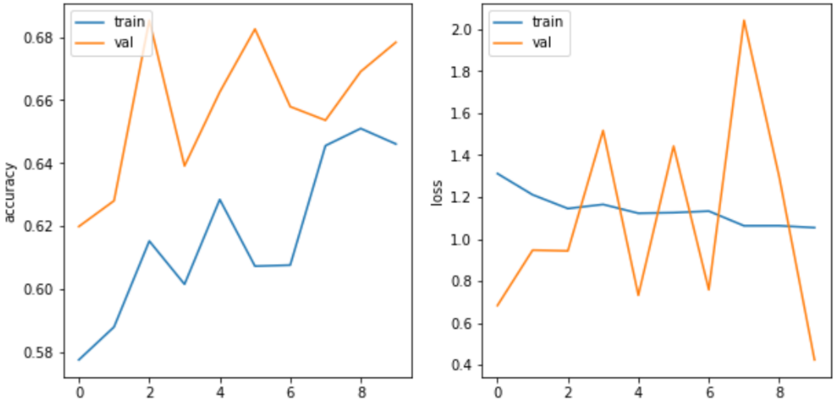


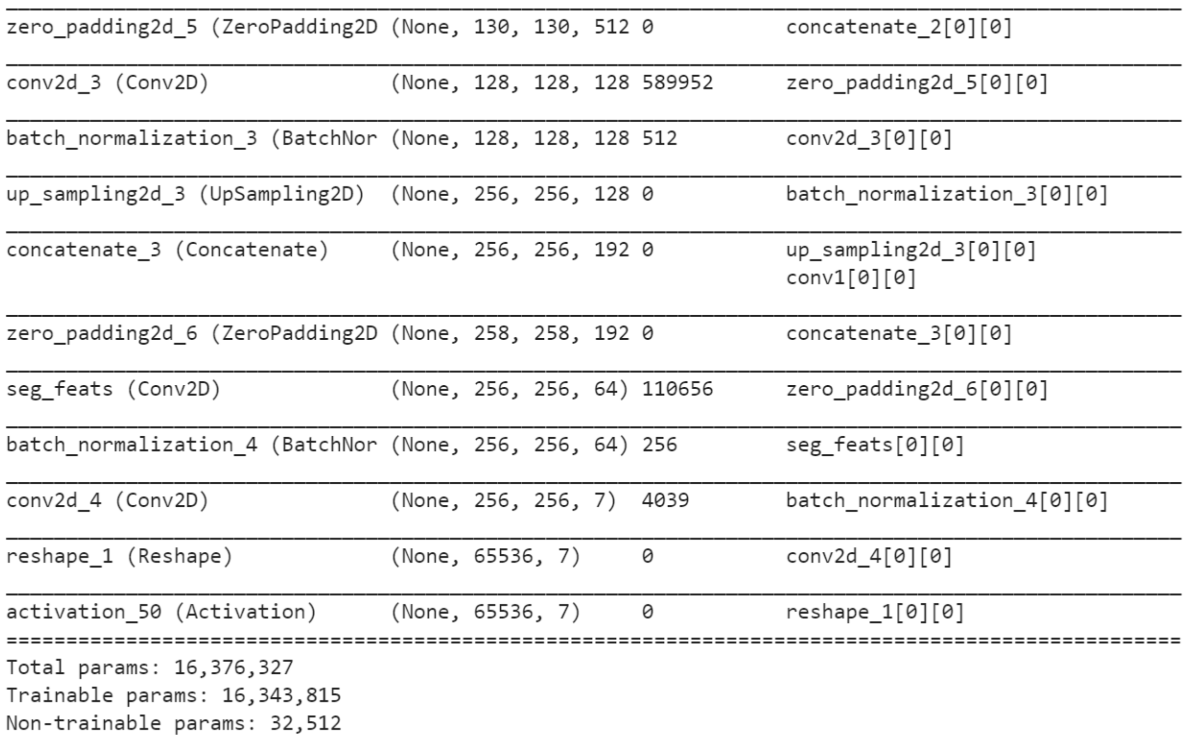


Land classification Models

- UNET
- VGG16_UNET
- RESNET50_UNET
- SEGNET
- VGG16_SEGNET
- RESNET50_SEGNET

VGG_UNET

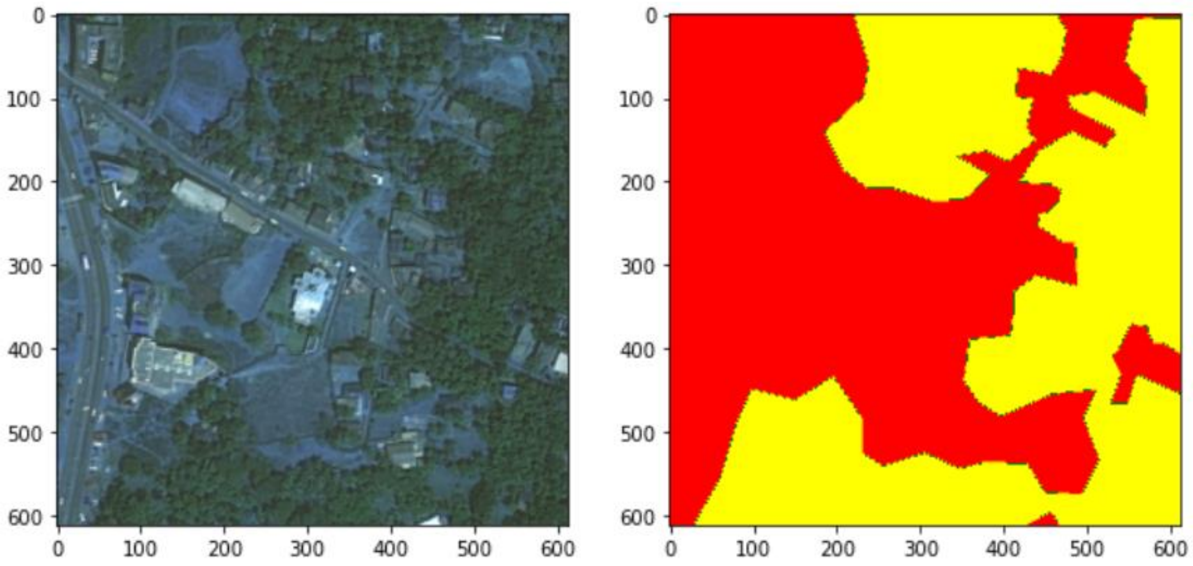


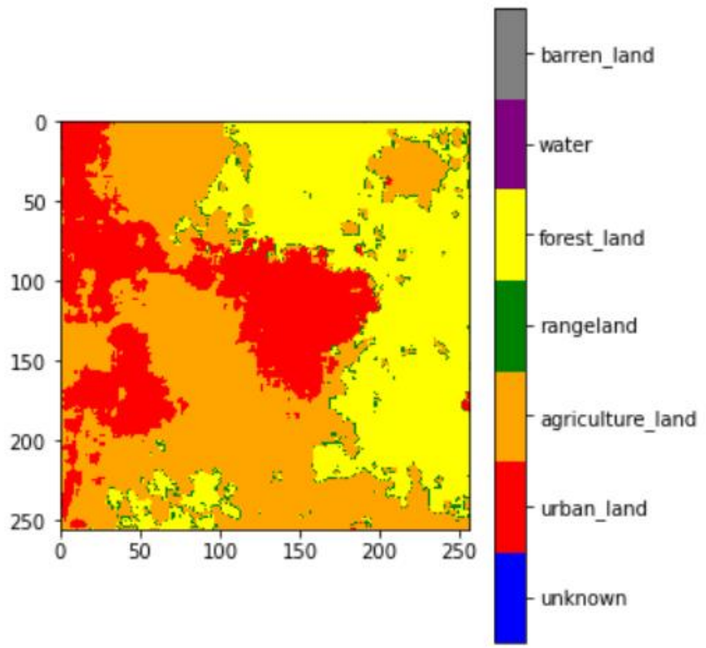


```
__________________________________________________________________________________________________
zero_padding2d_5 (ZeroPadding2D (None, 130, 130, 512 0           concatenate_2[0][0]
__________________________________________________________________________________________________
conv2d_3 (Conv2D)               (None, 128, 128, 128 589952      zero_padding2d_5[0][0]
__________________________________________________________________________________________________
batch_normalization_3 (BatchNor (None, 128, 128, 128 512         conv2d_3[0][0]
__________________________________________________________________________________________________
up_sampling2d_3 (UpSampling2D)  (None, 256, 256, 128 0           batch_normalization_3[0][0]
__________________________________________________________________________________________________
concatenate_3 (Concatenate)     (None, 256, 256, 192 0           up_sampling2d_3[0][0]
                                                                 conv1[0][0]
__________________________________________________________________________________________________
zero_padding2d_6 (ZeroPadding2D (None, 258, 258, 192 0           concatenate_3[0][0]
__________________________________________________________________________________________________
seg_feats (Conv2D)              (None, 256, 256, 64) 110656      zero_padding2d_6[0][0]
__________________________________________________________________________________________________
batch_normalization_4 (BatchNor (None, 256, 256, 64) 256         seg_feats[0][0]
__________________________________________________________________________________________________
conv2d_4 (Conv2D)               (None, 256, 256, 7)  4039        batch_normalization_4[0][0]
__________________________________________________________________________________________________
reshape_1 (Reshape)             (None, 65536, 7)     0           conv2d_4[0][0]
__________________________________________________________________________________________________
activation_50 (Activation)      (None, 65536, 7)     0           reshape_1[0][0]
==================================================================================================
Total params: 16,376,327
Trainable params: 16,343,815
Non-trainable params: 32,512
```

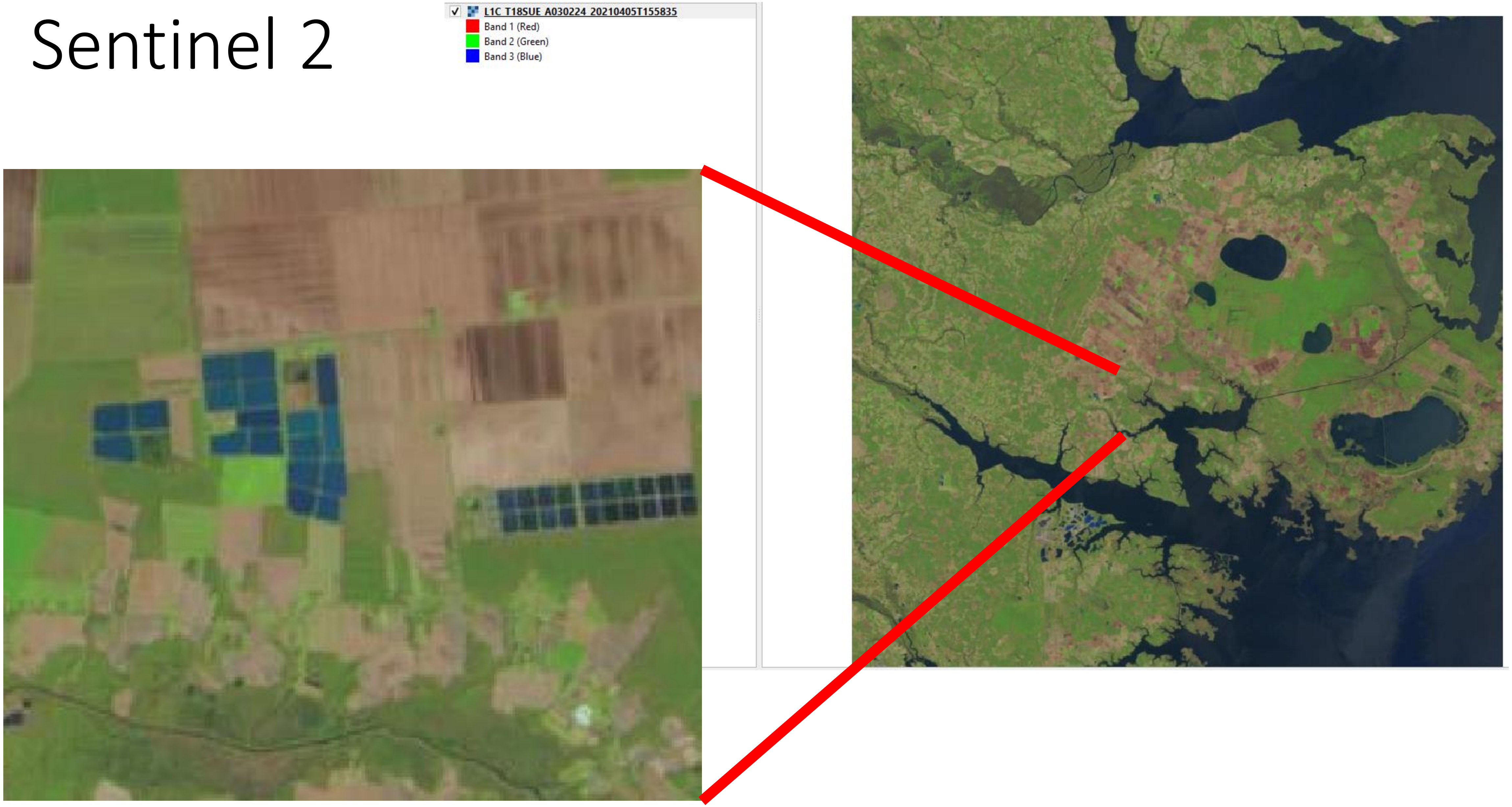
```
Epoch 00010: val_categorical_accuracy did not improve from 0.68539
```

```
Time Taken for testing: 0:40:19.564573
```





Sentinel 2



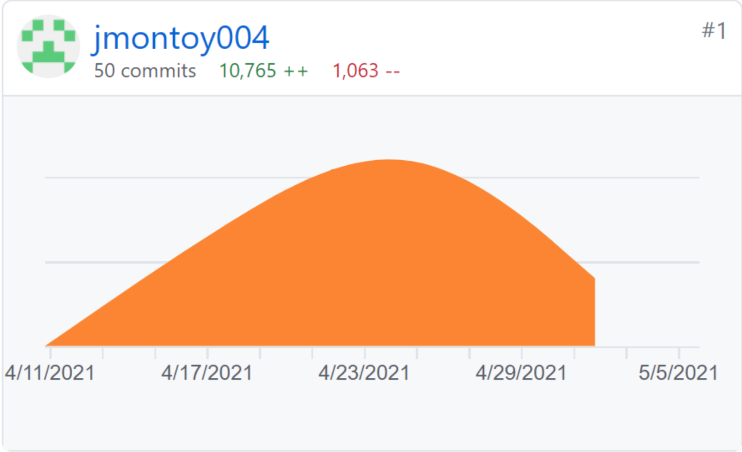
☑ L1C T18SUE A030224 20210405T155835

🟥 Band 1 (Red)

🟩 Band 2 (Green)

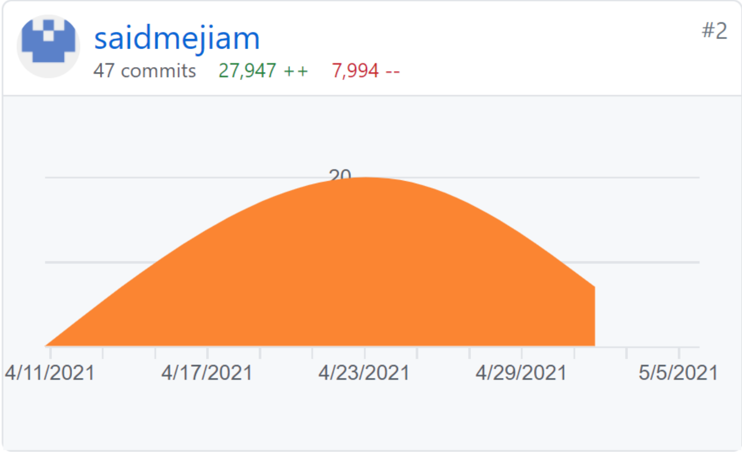
🟦 Band 3 (Blue)

Git hub commits 151 to 201 total



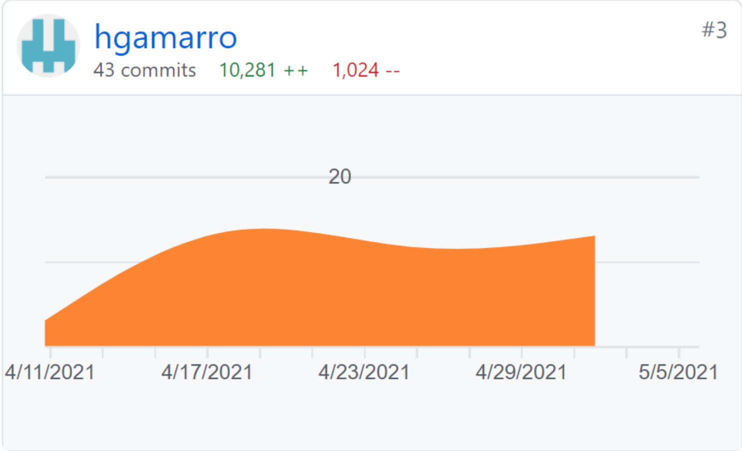
jmontoy004 #1

50 commits 10,765 ++ 1,063 --



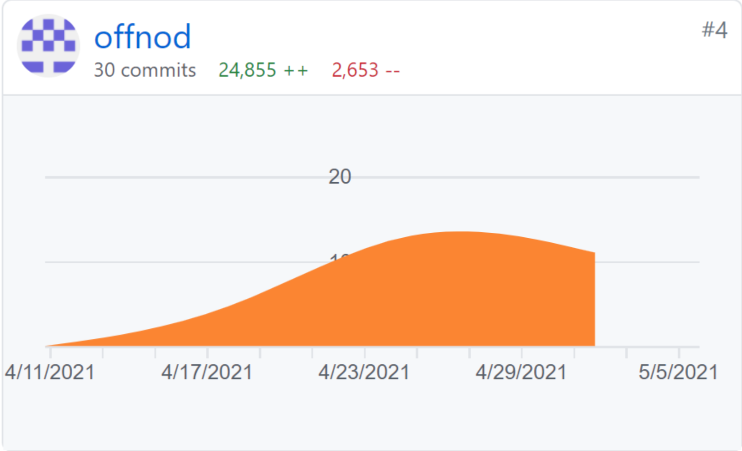
saidmejiam #2

47 commits 27,947 ++ 7,994 --



hgamarro #3

43 commits 10,281 ++ 1,024 --



offnod #4

30 commits 24,855 ++ 2,653 --

Notebook total

- HG: 4 to 10
- JB: 7 to 12
- JM: 4 to 10
- SM: 3 to 11