

```
In [1]: %load_ext autoreload
        %autoreload 2
```

```
In [2]: ##autoreload # When utils.py is updated
        from utils_unet_resunet import *
        from tensorflow.keras.preprocessing.image import ImageDataGenerator
        from model.models import Model_3
        from model.losses import WBCE
        root_path = 'imgs/'
```

```
In [3]: # Define data type
        img_type = 'FUSION'

        if img_type == 'FUSION':
            image_array = np.load(root_path+'New_Images/fus_stack.npy')

        if img_type == 'OPT':
            image_array = np.load(root_path+'New_Images/opt_stack.npy')

        if img_type == 'SAR':
            image_array = np.load(root_path+'New_Images/sar_stack.npy')
        print('Image stack:', image_array.shape)

        final_mask1 = np.load(root_path+'New_Images/'+'final_mask1.npy')
        print('Labels stack:', final_mask1.shape)

        h_, w_, channels = image_array.shape
        n_opt_layers = 20
```

Image stack: (10000, 7000, 24)  
Labels stack: (10000, 7000)

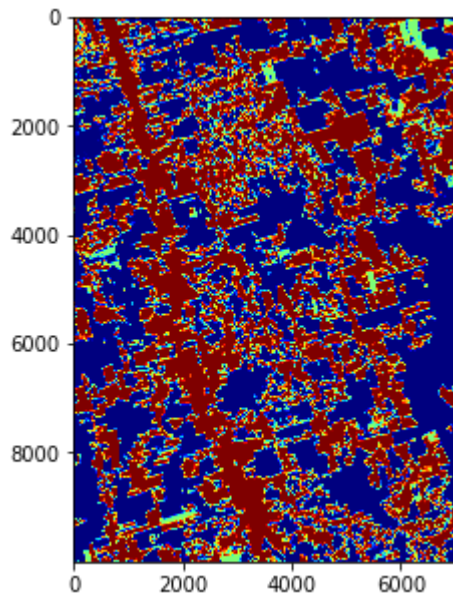
```
In [4]: # Create tile mask
        mask_tiles = create_mask(final_mask1.shape[0], final_mask1.shape[1], grid_size=(5, 4)
        image_array = image_array[:mask_tiles.shape[0], :mask_tiles.shape[1],:]
        final_mask1 = final_mask1[:mask_tiles.shape[0], :mask_tiles.shape[1]]

        print('mask: ', mask_tiles.shape)
        print('image stack: ', image_array.shape)
        print('ref :', final_mask1.shape)
        #plt.imshow(mask_tiles)
```

Tiles size: 2000 1750  
Mask size: (10000, 7000)  
mask: (10000, 7000)  
image stack: (10000, 7000, 24)  
ref : (10000, 7000)

```
In [5]: plt.figure(figsize=(10,5))
        plt.imshow(final_mask1, cmap = 'jet')
```

```
Out[5]: <matplotlib.image.AxesImage at 0x19461d7e0a0>
```



```
In [6]: # Define tiles for training, validation, and test sets
tiles_tr = [1,3,5,8,11,13,14,20]
tiles_val = [6,19]
tiles_ts = (list(set(np.arange(20)+1)-set(tiles_tr)-set(tiles_val)))

mask_tr_val = np.zeros((mask_tiles.shape)).astype('float32')
# Training and validation mask
for tr_ in tiles_tr:
    mask_tr_val[mask_tiles == tr_] = 1

for val_ in tiles_val:
    mask_tr_val[mask_tiles == val_] = 2

mask_amazon_ts = np.zeros((mask_tiles.shape)).astype('float32')
for ts_ in tiles_ts:
    mask_amazon_ts[mask_tiles == ts_] = 1
```

```
In [7]: # Create idx image to extract patches
overlap = 0.7
patch_size = 128
batch_size = 32
im_idx = create_idx_image(final_mask1)
patches_idx = extract_patches(im_idx, patch_size=(patch_size, patch_size), overlap=overlap)
patches_mask = extract_patches(mask_tr_val, patch_size=(patch_size, patch_size), overlap=overlap)
del im_idx
```

```
In [8]: # Selecting index trn val and test patches idx
idx_trn = np.squeeze(np.where(patches_mask.sum(axis=(1, 2))==patch_size**2))
idx_val = np.squeeze(np.where(patches_mask.sum(axis=(1, 2))==2*patch_size**2))
del patches_mask

patches_idx_trn = patches_idx[idx_trn]
patches_idx_val = patches_idx[idx_val]
del idx_trn, idx_val

print('Number of training patches: ', len(patches_idx_trn), 'Number of validation patches: ', len(patches_idx_val))
```

Number of training patches: 17110 Number of validation patches 4116

```
In [9]: # Extract patches with at least 2% of deforestation class
X_train = retrieve_idx_percentage(final_mask1, patches_idx_trn, patch_size, percentage=0.02)
```

```
X_valid = retrieve_idx_percentage(final_mask1, patches_idx_val, patch_size, pertenta
print(X_train.shape, X_valid.shape)
del patches_idx_trn, patches_idx_val
```

```
(1158, 128, 128) (341, 128, 128)
```

In [10]:

```
def batch_generator(batches, image, reference, target_size, number_class):
    """Take as input a Keras ImageGen (Iterator) and generate random
    crops from the image batches generated by the original iterator.
    """
    image = image.reshape(-1, image.shape[-1])
    reference = reference.reshape(final_mask1.shape[0]*final_mask1.shape[1])
    while True:
        batch_x, batch_y = next(batches)
        batch_x = np.squeeze(batch_x.astype('int64'))
        #print(batch_x.shape)
        batch_img = np.zeros((batch_x.shape[0], target_size, target_size, image.shap
        batch_ref = np.zeros((batch_x.shape[0], target_size, target_size, number_cla

        for i in range(batch_x.shape[0]):
            if np.random.rand()>0.5:
                batch_x[i] = np.rot90(batch_x[i], 1)
                batch_img[i] = image[batch_x[i]]
                batch_ref[i] = tf.keras.utils.to_categorical(reference[batch_x[i]] , num

        yield (batch_img, batch_ref)

train_datagen = ImageDataGenerator(horizontal_flip = True,
                                   vertical_flip = True)
valid_datagen = ImageDataGenerator(horizontal_flip = True,
                                   vertical_flip = True)

y_train = np.zeros((len(X_train)))
y_valid = np.zeros((len(X_valid)))

train_gen = train_datagen.flow(np.expand_dims(X_train, axis = -1), y_train,
                              batch_size=batch_size,
                              shuffle=True)

valid_gen = valid_datagen.flow(np.expand_dims(X_valid, axis = -1), y_valid,
                              batch_size=batch_size,
                              shuffle=False)

number_class = 3
train_gen_crops = batch_generator(train_gen, image_array, final_mask1, patch_size, n
valid_gen_crops = batch_generator(valid_gen, image_array, final_mask1, patch_size, n
```

In [11]:

```
exp = 3
path_exp = root_path+'experiments/exp'+str(exp)
path_models = path_exp+'/models'
path_maps = path_exp+'/pred_maps'

if not os.path.exists(path_exp):
    os.makedirs(path_exp)
if not os.path.exists(path_models):
    os.makedirs(path_models)
if not os.path.exists(path_maps):
    os.makedirs(path_maps)
```

In [12]:

```
# Define model
input_shape = (patch_size, patch_size, channels)
```

```

nb_filters = [32, 64, 128]

method = 'unet'
if method == 'unet':
    model = build_unet(input_shape, nb_filters, number_class)

if method == 'resnet':
    model = build_resnet(input_shape, nb_filters, number_class)

model = Model_3(nb_filters, number_class, n_opt_layers)

```

```

In [13]: # Parameters of the model
weights = [0.2, 0.8, 0]
adam = Adam(lr = 1e-4 , beta_1=0.9)
loss = weighted_categorical_crossentropy(weights)
#loss = WBCE(weights = weights)
#loss = WBCE(weights = weights, class_indexes = [0, 1])

```

```

In [14]: metrics_all = []
times=5
for tm in range(0,times):
    print('time: ', tm)

    rows = patch_size
    cols = patch_size
    adam = Adam(lr = 1e-4 , beta_1=0.9)

    loss = weighted_categorical_crossentropy(weights)
    #loss = WBCE(weights = weights)
    #loss = WBCE(weights = weights, class_indexes = [0, 1])

    #if method == 'unet':
    #    model = build_unet(input_shape, nb_filters, number_class)

    #if method == 'resnet':
    #    model = build_resnet(input_shape, nb_filters, number_class)

    model = Model_3(nb_filters, number_class, n_opt_layers)
    model.build((None,)+input_shape)

    model.compile(optimizer=adam, loss=loss, metrics=['accuracy'])
    model.summary()

    earllystop = EarlyStopping(monitor='val_loss', min_delta=0.0001, patience=10, ver
#earllystop = EarlyStopping(monitor='val_loss', min_delta=0.0001, patience=10, ve
#checkpoint = ModelCheckpoint(path_models+ '/' + method + '_' +str(tm)+'.h5', moni
checkpoint = ModelCheckpoint(path_models+ '/' + method + '_' +str(tm)+'.h5', monit
lr_reduce = ReduceLROnPlateau(factor=0.9, min_delta=0.0001, patience=5, verbose=
callbacks_list = [earllystop, checkpoint]
# train the model
start_training = time.time()
history = model.fit(train_gen_crops,
                    steps_per_epoch=len(X_train)*3//train_gen.batch_size,
                    validation_data=valid_gen_crops,
                    validation_steps=len(X_valid)*3//valid_gen.batch_size,
                    epochs=100,
                    callbacks=callbacks_list)
end_training = time.time() - start_training
metrics_all.append(end_training)
del model, history

```

time: 0

Model: "model\_3\_1"

Layer (type)	Output Shape	Param #
opt_encoder (UNET_Encoder)	multiple	540896
sar_encoder (UNET_Encoder)	multiple	536288
decoder (UNET_Decoder)	multiple	332000
opt_classifier (Classifier)	multiple	195
sar_classifier (Classifier)	multiple	195
fus_classifier (Classifier)	multiple	195
combination (CombinationLaye	multiple	3

Total params: 1,409,786

Trainable params: 1,409,769

Non-trainable params: 17

Epoch 1/100

108/108 [=====] - 27s 167ms/step - opt\_accuracy: 0.8090 - sar\_accuracy: 0.7333 - fus\_accuracy: 0.7946 - opt\_loss: 0.1012 - sar\_loss: 1.4144 - fusion\_loss: 0.1164 - loss: 1.6320 - val\_opt\_accuracy: 0.8550 - val\_sar\_accuracy: 0.7834 - val\_fus\_accuracy: 0.8562 - val\_opt\_loss: 0.1003 - val\_sar\_loss: 1.1737 - val\_fusion\_loss: 0.1063 - val\_loss: 1.3803

Epoch 00001: val\_loss improved from inf to 1.38034, saving model to imgs/experiments/exp3/models\unet\_0.h5

Epoch 2/100

108/108 [=====] - 15s 142ms/step - opt\_accuracy: 0.8630 - sar\_accuracy: 0.7597 - fus\_accuracy: 0.8532 - opt\_loss: 0.0776 - sar\_loss: 1.3153 - fusion\_loss: 0.0843 - loss: 1.4773 - val\_opt\_accuracy: 0.8603 - val\_sar\_accuracy: 0.7916 - val\_fus\_accuracy: 0.8621 - val\_opt\_loss: 0.1161 - val\_sar\_loss: 1.1364 - val\_fusion\_loss: 0.1364 - val\_loss: 1.3888

Epoch 00002: val\_loss did not improve from 1.38034

Epoch 3/100

108/108 [=====] - 15s 142ms/step - opt\_accuracy: 0.8766 - sar\_accuracy: 0.7655 - fus\_accuracy: 0.8686 - opt\_loss: 0.0707 - sar\_loss: 1.2715 - fusion\_loss: 0.0754 - loss: 1.4176 - val\_opt\_accuracy: 0.8644 - val\_sar\_accuracy: 0.7938 - val\_fus\_accuracy: 0.8661 - val\_opt\_loss: 0.0962 - val\_sar\_loss: 1.1654 - val\_fusion\_loss: 0.1062 - val\_loss: 1.3677

Epoch 00003: val\_loss improved from 1.38034 to 1.36774, saving model to imgs/experiments/exp3/models\unet\_0.h5

Epoch 4/100

108/108 [=====] - 15s 141ms/step - opt\_accuracy: 0.8833 - sar\_accuracy: 0.7714 - fus\_accuracy: 0.8765 - opt\_loss: 0.0668 - sar\_loss: 1.2420 - fusion\_loss: 0.0701 - loss: 1.3788 - val\_opt\_accuracy: 0.8662 - val\_sar\_accuracy: 0.7948 - val\_fus\_accuracy: 0.8677 - val\_opt\_loss: 0.0927 - val\_sar\_loss: 1.1487 - val\_fusion\_loss: 0.0962 - val\_loss: 1.3376

Epoch 00004: val\_loss improved from 1.36774 to 1.33762, saving model to imgs/experiments/exp3/models\unet\_0.h5

Epoch 5/100

108/108 [=====] - 15s 141ms/step - opt\_accuracy: 0.8892 - sar\_accuracy: 0.7843 - fus\_accuracy: 0.8843 - opt\_loss: 0.0633 - sar\_loss: 1.1742 - fusion\_loss: 0.0660 - loss: 1.3035 - val\_opt\_accuracy: 0.8622 - val\_sar\_accuracy: 0.8061 - val\_fus\_accuracy: 0.8636 - val\_opt\_loss: 0.1098 - val\_sar\_loss: 1.1421 - val\_fusion\_loss: 0.1137 - val\_loss: 1.3657

Epoch 00005: val\_loss did not improve from 1.33762

Epoch 6/100

108/108 [=====] - 15s 142ms/step - opt\_accuracy: 0.8881 - sar\_accuracy: 0.7874 - fus\_accuracy: 0.8857 - opt\_loss: 0.0627 - sar\_loss: 1.1627 - fusion\_loss: 0.0643 - loss: 1.2897 - val\_opt\_accuracy: 0.8609 - val\_sar\_accuracy: 0.7996 - val\_fus\_accuracy: 0.8616 - val\_opt\_loss: 0.1065 - val\_sar\_loss: 1.2520 - val\_fusion\_loss: 0.1107 - val\_loss: 1.4692

Epoch 00006: val\_loss did not improve from 1.33762

Epoch 7/100

108/108 [=====] - 15s 142ms/step - opt\_accuracy: 0.8927 - sar\_accuracy: 0.7906 - fus\_accuracy: 0.8903 - opt\_loss: 0.0597 - sar\_loss: 1.1307 - fusion\_loss: 0.0613 - loss: 1.2517 - val\_opt\_accuracy: 0.8597 - val\_sar\_accuracy: 0.8028 - val\_fus\_accuracy: 0.8600 - val\_opt\_loss: 0.1397 - val\_sar\_loss: 1.4114 - val\_fusion\_loss: 0.1484 - val\_loss: 1.6994

Epoch 00007: val\_loss did not improve from 1.33762

Epoch 8/100

108/108 [=====] - 15s 143ms/step - opt\_accuracy: 0.8987 - sar\_accuracy: 0.7944 - fus\_accuracy: 0.8964 - opt\_loss: 0.0548 - sar\_loss: 1.1189 - fusion\_loss: 0.0562 - loss: 1.2299 - val\_opt\_accuracy: 0.8508 - val\_sar\_accuracy: 0.8171 - val\_fus\_accuracy: 0.8573 - val\_opt\_loss: 0.1233 - val\_sar\_loss: 1.3483 - val\_fusion\_loss: 0.1293 - val\_loss: 1.6009

Epoch 00008: val\_loss did not improve from 1.33762

Epoch 9/100

108/108 [=====] - 15s 143ms/step - opt\_accuracy: 0.9006 - sar\_accuracy: 0.8019 - fus\_accuracy: 0.8982 - opt\_loss: 0.0524 - sar\_loss: 1.0690 - fusion\_loss: 0.0537 - loss: 1.1751 - val\_opt\_accuracy: 0.8587 - val\_sar\_accuracy: 0.8034 - val\_fus\_accuracy: 0.8593 - val\_opt\_loss: 0.1462 - val\_sar\_loss: 1.2925 - val\_fusion\_loss: 0.1508 - val\_loss: 1.5895

Epoch 00009: val\_loss did not improve from 1.33762

Epoch 10/100

108/108 [=====] - 16s 144ms/step - opt\_accuracy: 0.9080 - sar\_accuracy: 0.8131 - fus\_accuracy: 0.9058 - opt\_loss: 0.0471 - sar\_loss: 1.0184 - fusion\_loss: 0.0481 - loss: 1.1136 - val\_opt\_accuracy: 0.8569 - val\_sar\_accuracy: 0.8165 - val\_fus\_accuracy: 0.8560 - val\_opt\_loss: 0.1607 - val\_sar\_loss: 1.4087 - val\_fusion\_loss: 0.1687 - val\_loss: 1.7381

Epoch 00010: val\_loss did not improve from 1.33762

Epoch 11/100

108/108 [=====] - 15s 143ms/step - opt\_accuracy: 0.9129 - sar\_accuracy: 0.8159 - fus\_accuracy: 0.9108 - opt\_loss: 0.0433 - sar\_loss: 1.0005 - fusion\_loss: 0.0442 - loss: 1.0880 - val\_opt\_accuracy: 0.8548 - val\_sar\_accuracy: 0.8076 - val\_fus\_accuracy: 0.8539 - val\_opt\_loss: 0.1849 - val\_sar\_loss: 1.2210 - val\_fusion\_loss: 0.1853 - val\_loss: 1.5912

Epoch 00011: val\_loss did not improve from 1.33762

Epoch 12/100

108/108 [=====] - 16s 144ms/step - opt\_accuracy: 0.9146 - sar\_accuracy: 0.8122 - fus\_accuracy: 0.9126 - opt\_loss: 0.0419 - sar\_loss: 1.0164 - fusion\_loss: 0.0427 - loss: 1.1010 - val\_opt\_accuracy: 0.8554 - val\_sar\_accuracy: 0.8189 - val\_fus\_accuracy: 0.8564 - val\_opt\_loss: 0.1631 - val\_sar\_loss: 1.4456 - val\_fusion\_loss: 0.1645 - val\_loss: 1.7731

Epoch 00012: val\_loss did not improve from 1.33762

Epoch 13/100

108/108 [=====] - 16s 145ms/step - opt\_accuracy: 0.9160 - sar\_accuracy: 0.8161 - fus\_accuracy: 0.9142 - opt\_loss: 0.0406 - sar\_loss: 1.0051 - fusion\_loss: 0.0412 - loss: 1.0870 - val\_opt\_accuracy: 0.8622 - val\_sar\_accuracy: 0.7970 - val\_fus\_accuracy: 0.8616 - val\_opt\_loss: 0.1572 - val\_sar\_loss: 1.2666 - val\_fusion\_loss: 0.1576 - val\_loss: 1.5814

Epoch 00013: val\_loss did not improve from 1.33762

Epoch 14/100

108/108 [=====] - 16s 144ms/step - opt\_accuracy: 0.9174 - sar\_accuracy: 0.8097 - fus\_accuracy: 0.9158 - opt\_loss: 0.0399 - sar\_loss: 1.0278 - fusion\_loss: 0.0404 - loss: 1.1081 - val\_opt\_accuracy: 0.8639 - val\_sar\_accuracy: 0.8099 - val\_fus\_accuracy: 0.8646 - val\_opt\_loss: 0.2041 - val\_sar\_loss: 1.2658 - val\_fusion\_loss: 0.2037 - val\_loss: 1.6735

Epoch 00014: val\_loss did not improve from 1.33762

Epoch 00014: early stopping

time: 1

Model: "model\_3\_2"

Layer (type)	Output Shape	Param #
=====		
opt_encoder (UNET_Encoder)	multiple	540896
=====		
sar_encoder (UNET_Encoder)	multiple	536288
=====		
decoder (UNET_Decoder)	multiple	332000
=====		
opt_classifier (Classifier)	multiple	195
=====		
sar_classifier (Classifier)	multiple	195
=====		
fus_classifier (Classifier)	multiple	195
=====		
combination (CombinationLaye	multiple	3
=====		

Total params: 1,409,786

Trainable params: 1,409,769

Non-trainable params: 17

Epoch 1/100

108/108 [=====] - 18s 147ms/step - opt\_accuracy: 0.8364 - sar\_accuracy: 0.7471 - fus\_accuracy: 0.8148 - opt\_loss: 0.0967 - sar\_loss: 1.3514 - fusion\_loss: 0.1059 - loss: 1.5540 - val\_opt\_accuracy: 0.8444 - val\_sar\_accuracy: 0.7937 - val\_fus\_accuracy: 0.8561 - val\_opt\_loss: 0.1128 - val\_sar\_loss: 1.1922 - val\_fusion\_loss: 0.1113 - val\_loss: 1.4163

Epoch 00001: val\_loss improved from inf to 1.41631, saving model to imgs/experiments/exp3/models\unet\_1.h5

Epoch 2/100

108/108 [=====] - 16s 144ms/step - opt\_accuracy: 0.8673 - sar\_accuracy: 0.7753 - fus\_accuracy: 0.8605 - opt\_loss: 0.0783 - sar\_loss: 1.2160 - fusion\_loss: 0.0814 - loss: 1.3756 - val\_opt\_accuracy: 0.8409 - val\_sar\_accuracy: 0.8072 - val\_fus\_accuracy: 0.8485 - val\_opt\_loss: 0.1141 - val\_sar\_loss: 1.0818 - val\_fusion\_loss: 0.1070 - val\_loss: 1.3029

Epoch 00002: val\_loss improved from 1.41631 to 1.30294, saving model to imgs/experiments/exp3/models\unet\_1.h5

Epoch 3/100

108/108 [=====] - 16s 145ms/step - opt\_accuracy: 0.8795 - sar\_accuracy: 0.7858 - fus\_accuracy: 0.8740 - opt\_loss: 0.0693 - sar\_loss: 1.1536 - fusion\_loss: 0.0716 - loss: 1.2946 - val\_opt\_accuracy: 0.8463 - val\_sar\_accuracy: 0.8066 - val\_fus\_accuracy: 0.8504 - val\_opt\_loss: 0.1087 - val\_sar\_loss: 1.1100 - val\_fusion\_loss: 0.1042 - val\_loss: 1.3228

Epoch 00003: val\_loss did not improve from 1.30294

Epoch 4/100

108/108 [=====] - 16s 145ms/step - opt\_accuracy: 0.8870 - sar\_accuracy: 0.7880 - fus\_accuracy: 0.8815 - opt\_loss: 0.0642 - sar\_loss: 1.1224 - fusion\_loss: 0.0663 - loss: 1.2529 - val\_opt\_accuracy: 0.8605 - val\_sar\_accuracy: 0.8122 - val\_fus\_accuracy: 0.8619 - val\_opt\_loss: 0.1129 - val\_sar\_loss: 1.3904 - val\_fusion\_loss: 0.1207 - val\_loss: 1.6240

Epoch 00004: val\_loss did not improve from 1.30294

Epoch 5/100

108/108 [=====] - 16s 145ms/step - opt\_accuracy: 0.8927 - sar\_accuracy: 0.7941 - fus\_accuracy: 0.8881 - opt\_loss: 0.0595 - sar\_loss: 1.0959 - fusion\_loss: 0.0613 - loss: 1.2167 - val\_opt\_accuracy: 0.8583 - val\_sar\_accuracy: 0.8082 - val\_fus\_accuracy: 0.8603 - val\_opt\_loss: 0.1211 - val\_sar\_loss: 1.2198 - val\_fusion\_loss: 0.1197 - val\_loss: 1.4605

Epoch 00005: val\_loss did not improve from 1.30294

Epoch 6/100

108/108 [=====] - 16s 145ms/step - opt\_accuracy: 0.8992 - sar\_accuracy: 0.7964 - fus\_accuracy: 0.8949 - opt\_loss: 0.0540 - sar\_loss: 1.0845 - fusion\_loss: 0.0554 - loss: 1.1940 - val\_opt\_accuracy: 0.8666 - val\_sar\_accuracy: 0.8166 - val\_fus\_accuracy: 0.8649 - val\_opt\_loss: 0.1300 - val\_sar\_loss: 1.4995 - val\_fusion\_loss: 0.1433 - val\_loss: 1.7728

Epoch 00006: val\_loss did not improve from 1.30294

Epoch 7/100

108/108 [=====] - 16s 146ms/step - opt\_accuracy: 0.9006 - sar\_accuracy: 0.8022 - fus\_accuracy: 0.8972 - opt\_loss: 0.0523 - sar\_loss: 1.0552 - fusion\_loss: 0.0533 - loss: 1.1609 - val\_opt\_accuracy: 0.8501 - val\_sar\_accuracy: 0.8161 - val\_fus\_accuracy: 0.8542 - val\_opt\_loss: 0.1470 - val\_sar\_loss: 1.3835 - val\_fusion\_loss: 0.1481 - val\_loss: 1.6786

Epoch 00007: val\_loss did not improve from 1.30294

Epoch 8/100

108/108 [=====] - 16s 147ms/step - opt\_accuracy: 0.9043 - sar\_accuracy: 0.7953 - fus\_accuracy: 0.9013 - opt\_loss: 0.0495 - sar\_loss: 1.0883 - fusion\_loss: 0.0503 - loss: 1.1881 - val\_opt\_accuracy: 0.8640 - val\_sar\_accuracy: 0.8139 - val\_fus\_accuracy: 0.8643 - val\_opt\_loss: 0.1722 - val\_sar\_loss: 1.4797 - val\_fusion\_loss: 0.1768 - val\_loss: 1.8287

Epoch 00008: val\_loss did not improve from 1.30294

Epoch 9/100

108/108 [=====] - 16s 148ms/step - opt\_accuracy: 0.9113 - sar\_accuracy: 0.8089 - fus\_accuracy: 0.9089 - opt\_loss: 0.0436 - sar\_loss: 1.0128 - fusion\_loss: 0.0443 - loss: 1.1007 - val\_opt\_accuracy: 0.8641 - val\_sar\_accuracy: 0.8111 - val\_fus\_accuracy: 0.8650 - val\_opt\_loss: 0.1735 - val\_sar\_loss: 1.4068 - val\_fusion\_loss: 0.1725 - val\_loss: 1.7528

Epoch 00009: val\_loss did not improve from 1.30294

Epoch 10/100

108/108 [=====] - 16s 148ms/step - opt\_accuracy: 0.9086 - sar\_accuracy: 0.8013 - fus\_accuracy: 0.9063 - opt\_loss: 0.0454 - sar\_loss: 1.0837 - fusion\_loss: 0.0460 - loss: 1.1751 - val\_opt\_accuracy: 0.8659 - val\_sar\_accuracy: 0.8109 - val\_fus\_accuracy: 0.8661 - val\_opt\_loss: 0.2097 - val\_sar\_loss: 1.4906 - val\_fusion\_loss: 0.2101 - val\_loss: 1.9103

Epoch 00010: val\_loss did not improve from 1.30294

Epoch 11/100

108/108 [=====] - 16s 148ms/step - opt\_accuracy: 0.9171 - sar\_accuracy: 0.8070 - fus\_accuracy: 0.9152 - opt\_loss: 0.0387 - sar\_loss: 1.0601 - fusion\_loss: 0.0393 - loss: 1.1382 - val\_opt\_accuracy: 0.8701 - val\_sar\_accuracy: 0.8047 - val\_fus\_accuracy: 0.8661 - val\_opt\_loss: 0.2088 - val\_sar\_loss: 1.3488 - val\_fusion\_loss: 0.2057 - val\_loss: 1.7634

Epoch 00011: val\_loss did not improve from 1.30294

Epoch 12/100

108/108 [=====] - 16s 149ms/step - opt\_accuracy: 0.9185 - sar\_accuracy: 0.8087 - fus\_accuracy: 0.9166 - opt\_loss: 0.0378 - sar\_loss: 1.0360 - fusion\_loss: 0.0383 - loss: 1.1121 - val\_opt\_accuracy: 0.8684 - val\_sar\_accuracy: 0.8067 - val\_fus\_accuracy: 0.8686 - val\_opt\_loss: 0.1610 - val\_sar\_loss: 1.1294 - val\_fusion\_loss: 0.1578 - val\_loss: 1.4483

Epoch 00012: val\_loss did not improve from 1.30294

Epoch 00012: early stopping



time: 2  
Model: "model\_3\_3"

Layer (type)	Output Shape	Param #
opt_encoder (UNET_Encoder)	multiple	540896
sar_encoder (UNET_Encoder)	multiple	536288
decoder (UNET_Decoder)	multiple	332000
opt_classifier (Classifier)	multiple	195
sar_classifier (Classifier)	multiple	195
fus_classifier (Classifier)	multiple	195
combination (CombinationLaye	multiple	3
Total params: 1,409,786		
Trainable params: 1,409,769		
Non-trainable params: 17		

Epoch 1/100

108/108 [=====] - 18s 152ms/step - opt\_accuracy: 0.8308 - sar\_accuracy: 0.7375 - fus\_accuracy: 0.8136 - opt\_loss: 0.0966 - sar\_loss: 1.4138 - fusion\_loss: 0.1073 - loss: 1.6178 - val\_opt\_accuracy: 0.8645 - val\_sar\_accuracy: 0.7634 - val\_fus\_accuracy: 0.8633 - val\_opt\_loss: 0.1166 - val\_sar\_loss: 1.2766 - val\_fusion\_loss: 0.1052 - val\_loss: 1.4984

Epoch 00001: val\_loss improved from inf to 1.49842, saving model to imgs/experiments/exp3/models\unet\_2.h5

Epoch 2/100

108/108 [=====] - 16s 149ms/step - opt\_accuracy: 0.8683 - sar\_accuracy: 0.7570 - fus\_accuracy: 0.8561 - opt\_loss: 0.0766 - sar\_loss: 1.3320 - fusion\_loss: 0.0818 - loss: 1.4905 - val\_opt\_accuracy: 0.8525 - val\_sar\_accuracy: 0.7905 - val\_fus\_accuracy: 0.8572 - val\_opt\_loss: 0.0973 - val\_sar\_loss: 1.1649 - val\_fusion\_loss: 0.0911 - val\_loss: 1.3533

Epoch 00002: val\_loss improved from 1.49842 to 1.35325, saving model to imgs/experiments/exp3/models\unet\_2.h5

Epoch 3/100

108/108 [=====] - 16s 149ms/step - opt\_accuracy: 0.8782 - sar\_accuracy: 0.7716 - fus\_accuracy: 0.8696 - opt\_loss: 0.0685 - sar\_loss: 1.2704 - fusion\_loss: 0.0717 - loss: 1.4106 - val\_opt\_accuracy: 0.8588 - val\_sar\_accuracy: 0.7966 - val\_fus\_accuracy: 0.8593 - val\_opt\_loss: 0.1413 - val\_sar\_loss: 1.1403 - val\_fusion\_loss: 0.1327 - val\_loss: 1.4143

Epoch 00003: val\_loss did not improve from 1.35325

Epoch 4/100

108/108 [=====] - 16s 149ms/step - opt\_accuracy: 0.8859 - sar\_accuracy: 0.7710 - fus\_accuracy: 0.8797 - opt\_loss: 0.0625 - sar\_loss: 1.2405 - fusion\_loss: 0.0647 - loss: 1.3678 - val\_opt\_accuracy: 0.8627 - val\_sar\_accuracy: 0.7947 - val\_fus\_accuracy: 0.8638 - val\_opt\_loss: 0.1424 - val\_sar\_loss: 1.1968 - val\_fusion\_loss: 0.1374 - val\_loss: 1.4766

Epoch 00004: val\_loss did not improve from 1.35325

Epoch 5/100

108/108 [=====] - 16s 149ms/step - opt\_accuracy: 0.8912 - sar\_accuracy: 0.7581 - fus\_accuracy: 0.8874 - opt\_loss: 0.0598 - sar\_loss: 1.3032 - fusion\_loss: 0.0612 - loss: 1.4242 - val\_opt\_accuracy: 0.8534 - val\_sar\_accuracy: 0.7949 - val\_fus\_accuracy: 0.8565 - val\_opt\_loss: 0.1540 - val\_sar\_loss: 1.2306 - val\_fusion\_loss: 0.1519 - val\_loss: 1.5364

Epoch 00005: val\_loss did not improve from 1.35325

Epoch 6/100

108/108 [=====] - 16s 150ms/step - opt\_accuracy: 0.8946 - sar\_accuracy: 0.7481 - fus\_accuracy: 0.8923 - opt\_loss: 0.0560 - sar\_loss: 1.3224 - fusion\_loss: 0.0571 - loss: 1.4355 - val\_opt\_accuracy: 0.8522 - val\_sar\_accuracy: 0.7856 - val\_fus\_accuracy: 0.8548 - val\_opt\_loss: 0.1801 - val\_sar\_loss: 1.2529 - val\_fusion\_loss: 0.1769 - val\_loss: 1.6099

Epoch 00006: val\_loss did not improve from 1.35325

Epoch 7/100

108/108 [=====] - 16s 150ms/step - opt\_accuracy: 0.8989 - sar\_accuracy: 0.7639 - fus\_accuracy: 0.8967 - opt\_loss: 0.0530 - sar\_loss: 1.2578 - fusion\_loss: 0.0542 - loss: 1.3650 - val\_opt\_accuracy: 0.8601 - val\_sar\_accuracy: 0.7788 - val\_fus\_accuracy: 0.8614 - val\_opt\_loss: 0.1539 - val\_sar\_loss: 1.2935 - val\_fusion\_loss: 0.1489 - val\_loss: 1.5963

Epoch 00007: val\_loss did not improve from 1.35325

Epoch 8/100

108/108 [=====] - 16s 150ms/step - opt\_accuracy: 0.9039 - sar\_accuracy: 0.7542 - fus\_accuracy: 0.9016 - opt\_loss: 0.0487 - sar\_loss: 1.2756 - fusion\_loss: 0.0498 - loss: 1.3740 - val\_opt\_accuracy: 0.8599 - val\_sar\_accuracy: 0.7740 - val\_fus\_accuracy: 0.8601 - val\_opt\_loss: 0.1525 - val\_sar\_loss: 1.1765 - val\_fusion\_loss: 0.1458 - val\_loss: 1.4748

Epoch 00008: val\_loss did not improve from 1.35325

Epoch 9/100

108/108 [=====] - 16s 151ms/step - opt\_accuracy: 0.9090 - sar\_accuracy: 0.7455 - fus\_accuracy: 0.9071 - opt\_loss: 0.0444 - sar\_loss: 1.3345 - fusion\_loss: 0.0452 - loss: 1.4240 - val\_opt\_accuracy: 0.8634 - val\_sar\_accuracy: 0.7973 - val\_fus\_accuracy: 0.8636 - val\_opt\_loss: 0.1749 - val\_sar\_loss: 1.2772 - val\_fusion\_loss: 0.1688 - val\_loss: 1.6208

Epoch 00009: val\_loss did not improve from 1.35325

Epoch 10/100

108/108 [=====] - 16s 152ms/step - opt\_accuracy: 0.9135 - sar\_accuracy: 0.7497 - fus\_accuracy: 0.9118 - opt\_loss: 0.0410 - sar\_loss: 1.3273 - fusion\_loss: 0.0418 - loss: 1.4100 - val\_opt\_accuracy: 0.8709 - val\_sar\_accuracy: 0.7213 - val\_fus\_accuracy: 0.8711 - val\_opt\_loss: 0.1456 - val\_sar\_loss: 1.2655 - val\_fusion\_loss: 0.1423 - val\_loss: 1.5534

Epoch 00010: val\_loss did not improve from 1.35325

Epoch 11/100

108/108 [=====] - 16s 152ms/step - opt\_accuracy: 0.9148 - sar\_accuracy: 0.7294 - fus\_accuracy: 0.9132 - opt\_loss: 0.0396 - sar\_loss: 1.3891 - fusion\_loss: 0.0402 - loss: 1.4690 - val\_opt\_accuracy: 0.8644 - val\_sar\_accuracy: 0.7421 - val\_fus\_accuracy: 0.8645 - val\_opt\_loss: 0.1674 - val\_sar\_loss: 1.1989 - val\_fusion\_loss: 0.1653 - val\_loss: 1.5316

Epoch 00011: val\_loss did not improve from 1.35325

Epoch 12/100

108/108 [=====] - 16s 153ms/step - opt\_accuracy: 0.9192 - sar\_accuracy: 0.7342 - fus\_accuracy: 0.9179 - opt\_loss: 0.0363 - sar\_loss: 1.3837 - fusion\_loss: 0.0367 - loss: 1.4567 - val\_opt\_accuracy: 0.8659 - val\_sar\_accuracy: 0.7460 - val\_fus\_accuracy: 0.8657 - val\_opt\_loss: 0.1909 - val\_sar\_loss: 1.1940 - val\_fusion\_loss: 0.1892 - val\_loss: 1.5741

Epoch 00012: val\_loss did not improve from 1.35325

Epoch 00012: early stopping

time: 3

Model: "model\_3\_4"

Layer (type)	Output Shape	Param #
=====		
opt_encoder (UNET_Encoder)	multiple	540896
=====		
sar_encoder (UNET_Encoder)	multiple	536288

decoder (UNET_Decoder)	multiple	332000
opt_classifier (Classifier)	multiple	195
sar_classifier (Classifier)	multiple	195
fus_classifier (Classifier)	multiple	195
combination (CombinationLaye	multiple	3

=====

Total params: 1,409,786  
 Trainable params: 1,409,769  
 Non-trainable params: 17

Epoch 1/100

108/108 [=====] - 19s 159ms/step - opt\_accuracy: 0.8115 - s  
 ar\_accuracy: 0.7403 - fus\_accuracy: 0.8058 - opt\_loss: 0.1036 - sar\_loss: 1.4148 - f  
 usion\_loss: 0.1058 - loss: 1.6242 - val\_opt\_accuracy: 0.8606 - val\_sar\_accuracy: 0.7  
 880 - val\_fus\_accuracy: 0.8600 - val\_opt\_loss: 0.0965 - val\_sar\_loss: 1.1901 - val\_f  
 usion\_loss: 0.0991 - val\_loss: 1.3856

Epoch 00001: val\_loss improved from inf to 1.38559, saving model to imgs/experiment  
 s/exp3/models\unet\_3.h5

Epoch 2/100

108/108 [=====] - 17s 154ms/step - opt\_accuracy: 0.8700 - s  
 ar\_accuracy: 0.7669 - fus\_accuracy: 0.8630 - opt\_loss: 0.0775 - sar\_loss: 1.3103 - f  
 usion\_loss: 0.0791 - loss: 1.4669 - val\_opt\_accuracy: 0.8632 - val\_sar\_accuracy: 0.7  
 970 - val\_fus\_accuracy: 0.8640 - val\_opt\_loss: 0.1022 - val\_sar\_loss: 1.1727 - val\_f  
 usion\_loss: 0.1011 - val\_loss: 1.3760

Epoch 00002: val\_loss improved from 1.38559 to 1.37596, saving model to imgs/experim  
 ents/exp3/models\unet\_3.h5

Epoch 3/100

108/108 [=====] - 17s 155ms/step - opt\_accuracy: 0.8799 - s  
 ar\_accuracy: 0.7866 - fus\_accuracy: 0.8744 - opt\_loss: 0.0696 - sar\_loss: 1.1877 - f  
 usion\_loss: 0.0714 - loss: 1.3287 - val\_opt\_accuracy: 0.8602 - val\_sar\_accuracy: 0.8  
 069 - val\_fus\_accuracy: 0.8610 - val\_opt\_loss: 0.1004 - val\_sar\_loss: 1.1717 - val\_f  
 usion\_loss: 0.1033 - val\_loss: 1.3754

Epoch 00003: val\_loss improved from 1.37596 to 1.37542, saving model to imgs/experim  
 ents/exp3/models\unet\_3.h5

Epoch 4/100

108/108 [=====] - 17s 154ms/step - opt\_accuracy: 0.8852 - s  
 ar\_accuracy: 0.7947 - fus\_accuracy: 0.8801 - opt\_loss: 0.0656 - sar\_loss: 1.1440 - f  
 usion\_loss: 0.0673 - loss: 1.2769 - val\_opt\_accuracy: 0.8563 - val\_sar\_accuracy: 0.8  
 080 - val\_fus\_accuracy: 0.8562 - val\_opt\_loss: 0.1210 - val\_sar\_loss: 1.1671 - val\_f  
 usion\_loss: 0.1212 - val\_loss: 1.4093

Epoch 00004: val\_loss did not improve from 1.37542

Epoch 5/100

108/108 [=====] - 17s 156ms/step - opt\_accuracy: 0.8915 - s  
 ar\_accuracy: 0.7975 - fus\_accuracy: 0.8868 - opt\_loss: 0.0609 - sar\_loss: 1.1235 - f  
 usion\_loss: 0.0625 - loss: 1.2469 - val\_opt\_accuracy: 0.8596 - val\_sar\_accuracy: 0.7  
 948 - val\_fus\_accuracy: 0.8574 - val\_opt\_loss: 0.1412 - val\_sar\_loss: 1.1347 - val\_f  
 usion\_loss: 0.1341 - val\_loss: 1.4100

Epoch 00005: val\_loss did not improve from 1.37542

Epoch 6/100

108/108 [=====] - 17s 156ms/step - opt\_accuracy: 0.8950 - s  
 ar\_accuracy: 0.8039 - fus\_accuracy: 0.8923 - opt\_loss: 0.0573 - sar\_loss: 1.0807 - f  
 usion\_loss: 0.0582 - loss: 1.1962 - val\_opt\_accuracy: 0.8607 - val\_sar\_accuracy: 0.8  
 156 - val\_fus\_accuracy: 0.8607 - val\_opt\_loss: 0.1250 - val\_sar\_loss: 1.1898 - val\_f  
 usion\_loss: 0.1206 - val\_loss: 1.4354

Epoch 00006: val\_loss did not improve from 1.37542

Epoch 7/100

108/108 [=====] - 17s 156ms/step - opt\_accuracy: 0.9013 - sar\_accuracy: 0.8129 - fus\_accuracy: 0.8996 - opt\_loss: 0.0525 - sar\_loss: 1.0554 - fusion\_loss: 0.0531 - loss: 1.1611 - val\_opt\_accuracy: 0.8627 - val\_sar\_accuracy: 0.8185 - val\_fus\_accuracy: 0.8627 - val\_opt\_loss: 0.1510 - val\_sar\_loss: 1.2455 - val\_fusion\_loss: 0.1478 - val\_loss: 1.5442

Epoch 00007: val\_loss did not improve from 1.37542

Epoch 8/100

108/108 [=====] - 17s 157ms/step - opt\_accuracy: 0.9071 - sar\_accuracy: 0.8190 - fus\_accuracy: 0.9057 - opt\_loss: 0.0471 - sar\_loss: 0.9974 - fusion\_loss: 0.0476 - loss: 1.0921 - val\_opt\_accuracy: 0.8631 - val\_sar\_accuracy: 0.8190 - val\_fus\_accuracy: 0.8637 - val\_opt\_loss: 0.1562 - val\_sar\_loss: 1.4406 - val\_fusion\_loss: 0.1592 - val\_loss: 1.7561

Epoch 00008: val\_loss did not improve from 1.37542

Epoch 9/100

108/108 [=====] - 17s 157ms/step - opt\_accuracy: 0.9055 - sar\_accuracy: 0.8170 - fus\_accuracy: 0.9052 - opt\_loss: 0.0476 - sar\_loss: 1.0137 - fusion\_loss: 0.0478 - loss: 1.1092 - val\_opt\_accuracy: 0.8668 - val\_sar\_accuracy: 0.7915 - val\_fus\_accuracy: 0.8620 - val\_opt\_loss: 0.1430 - val\_sar\_loss: 1.2370 - val\_fusion\_loss: 0.1390 - val\_loss: 1.5190

Epoch 00009: val\_loss did not improve from 1.37542

Epoch 10/100

108/108 [=====] - 17s 157ms/step - opt\_accuracy: 0.9140 - sar\_accuracy: 0.8197 - fus\_accuracy: 0.9134 - opt\_loss: 0.0412 - sar\_loss: 0.9979 - fusion\_loss: 0.0415 - loss: 1.0806 - val\_opt\_accuracy: 0.8675 - val\_sar\_accuracy: 0.8159 - val\_fus\_accuracy: 0.8671 - val\_opt\_loss: 0.1670 - val\_sar\_loss: 1.2672 - val\_fusion\_loss: 0.1653 - val\_loss: 1.5996

Epoch 00010: val\_loss did not improve from 1.37542

Epoch 11/100

108/108 [=====] - 17s 159ms/step - opt\_accuracy: 0.9154 - sar\_accuracy: 0.8167 - fus\_accuracy: 0.9149 - opt\_loss: 0.0398 - sar\_loss: 1.0114 - fusion\_loss: 0.0401 - loss: 1.0913 - val\_opt\_accuracy: 0.8616 - val\_sar\_accuracy: 0.8115 - val\_fus\_accuracy: 0.8630 - val\_opt\_loss: 0.1506 - val\_sar\_loss: 1.2449 - val\_fusion\_loss: 0.1510 - val\_loss: 1.5465

Epoch 00011: val\_loss did not improve from 1.37542

Epoch 12/100

108/108 [=====] - 17s 160ms/step - opt\_accuracy: 0.9188 - sar\_accuracy: 0.8157 - fus\_accuracy: 0.9184 - opt\_loss: 0.0370 - sar\_loss: 1.0227 - fusion\_loss: 0.0372 - loss: 1.0969 - val\_opt\_accuracy: 0.8551 - val\_sar\_accuracy: 0.7923 - val\_fus\_accuracy: 0.8568 - val\_opt\_loss: 0.1935 - val\_sar\_loss: 1.3134 - val\_fusion\_loss: 0.1891 - val\_loss: 1.6960

Epoch 00012: val\_loss did not improve from 1.37542

Epoch 13/100

108/108 [=====] - 17s 160ms/step - opt\_accuracy: 0.9187 - sar\_accuracy: 0.8217 - fus\_accuracy: 0.9185 - opt\_loss: 0.0369 - sar\_loss: 0.9789 - fusion\_loss: 0.0369 - loss: 1.0527 - val\_opt\_accuracy: 0.8614 - val\_sar\_accuracy: 0.8168 - val\_fus\_accuracy: 0.8595 - val\_opt\_loss: 0.2199 - val\_sar\_loss: 1.6635 - val\_fusion\_loss: 0.2188 - val\_loss: 2.1022

Epoch 00013: val\_loss did not improve from 1.37542

Epoch 00013: early stopping

time: 4

Model: "model\_3\_5"

Layer (type)	Output Shape	Param #
=====		
opt_encoder (UNET_Encoder)	multiple	540896

sar_encoder (UNET_Encoder)	multiple	536288
decoder (UNET_Decoder)	multiple	332000
opt_classifier (Classifier)	multiple	195
sar_classifier (Classifier)	multiple	195
fus_classifier (Classifier)	multiple	195
combination (CombinationLaye	multiple	3
=====		
Total params: 1,409,786		
Trainable params: 1,409,769		
Non-trainable params: 17		

Epoch 1/100

108/108 [=====] - 19s 163ms/step - opt\_accuracy: 0.8313 - sar\_accuracy: 0.7335 - fus\_accuracy: 0.8024 - opt\_loss: 0.1008 - sar\_loss: 1.4361 - fusion\_loss: 0.1124 - loss: 1.6493 - val\_opt\_accuracy: 0.8524 - val\_sar\_accuracy: 0.7714 - val\_fus\_accuracy: 0.8556 - val\_opt\_loss: 0.0978 - val\_sar\_loss: 1.2412 - val\_fusion\_loss: 0.0966 - val\_loss: 1.4356

Epoch 00001: val\_loss improved from inf to 1.43561, saving model to imgs/experiments/exp3/models\unet\_4.h5

Epoch 2/100

108/108 [=====] - 17s 161ms/step - opt\_accuracy: 0.8693 - sar\_accuracy: 0.7455 - fus\_accuracy: 0.8568 - opt\_loss: 0.0781 - sar\_loss: 1.3690 - fusion\_loss: 0.0836 - loss: 1.5308 - val\_opt\_accuracy: 0.8528 - val\_sar\_accuracy: 0.7714 - val\_fus\_accuracy: 0.8498 - val\_opt\_loss: 0.1217 - val\_sar\_loss: 1.1998 - val\_fusion\_loss: 0.1172 - val\_loss: 1.4387

Epoch 00002: val\_loss did not improve from 1.43561

Epoch 3/100

108/108 [=====] - 17s 161ms/step - opt\_accuracy: 0.8779 - sar\_accuracy: 0.7560 - fus\_accuracy: 0.8677 - opt\_loss: 0.0710 - sar\_loss: 1.3211 - fusion\_loss: 0.0760 - loss: 1.4681 - val\_opt\_accuracy: 0.8661 - val\_sar\_accuracy: 0.7934 - val\_fus\_accuracy: 0.8626 - val\_opt\_loss: 0.0998 - val\_sar\_loss: 1.1546 - val\_fusion\_loss: 0.1055 - val\_loss: 1.3599

Epoch 00003: val\_loss improved from 1.43561 to 1.35987, saving model to imgs/experiments/exp3/models\unet\_4.h5

Epoch 4/100

108/108 [=====] - 17s 161ms/step - opt\_accuracy: 0.8835 - sar\_accuracy: 0.7202 - fus\_accuracy: 0.8774 - opt\_loss: 0.0662 - sar\_loss: 1.4233 - fusion\_loss: 0.0698 - loss: 1.5592 - val\_opt\_accuracy: 0.8566 - val\_sar\_accuracy: 0.7469 - val\_fus\_accuracy: 0.8573 - val\_opt\_loss: 0.1235 - val\_sar\_loss: 1.2265 - val\_fusion\_loss: 0.1223 - val\_loss: 1.4723

Epoch 00004: val\_loss did not improve from 1.35987

Epoch 5/100

108/108 [=====] - 17s 162ms/step - opt\_accuracy: 0.8871 - sar\_accuracy: 0.7288 - fus\_accuracy: 0.8820 - opt\_loss: 0.0638 - sar\_loss: 1.4087 - fusion\_loss: 0.0663 - loss: 1.5388 - val\_opt\_accuracy: 0.8609 - val\_sar\_accuracy: 0.7510 - val\_fus\_accuracy: 0.8599 - val\_opt\_loss: 0.1268 - val\_sar\_loss: 1.2021 - val\_fusion\_loss: 0.1300 - val\_loss: 1.4589

Epoch 00005: val\_loss did not improve from 1.35987

Epoch 6/100

108/108 [=====] - 18s 163ms/step - opt\_accuracy: 0.8923 - sar\_accuracy: 0.7318 - fus\_accuracy: 0.8883 - opt\_loss: 0.0597 - sar\_loss: 1.4146 - fusion\_loss: 0.0616 - loss: 1.5359 - val\_opt\_accuracy: 0.8667 - val\_sar\_accuracy: 0.7858 - val\_fus\_accuracy: 0.8602 - val\_opt\_loss: 0.1357 - val\_sar\_loss: 1.1745 - val\_fusion\_loss: 0.1399 - val\_loss: 1.4501

Epoch 00006: val\_loss did not improve from 1.35987

Epoch 7/100

108/108 [=====] - 18s 163ms/step - opt\_accuracy: 0.8958 - sar\_accuracy: 0.7334 - fus\_accuracy: 0.8933 - opt\_loss: 0.0566 - sar\_loss: 1.4000 - fusion\_loss: 0.0577 - loss: 1.5143 - val\_opt\_accuracy: 0.8567 - val\_sar\_accuracy: 0.7528 - val\_fus\_accuracy: 0.8502 - val\_opt\_loss: 0.1028 - val\_sar\_loss: 1.1965 - val\_fusion\_loss: 0.1052 - val\_loss: 1.4046

Epoch 00007: val\_loss did not improve from 1.35987

Epoch 8/100

108/108 [=====] - 18s 163ms/step - opt\_accuracy: 0.9001 - sar\_accuracy: 0.7356 - fus\_accuracy: 0.8983 - opt\_loss: 0.0530 - sar\_loss: 1.3944 - fusion\_loss: 0.0539 - loss: 1.5013 - val\_opt\_accuracy: 0.8682 - val\_sar\_accuracy: 0.7553 - val\_fus\_accuracy: 0.8679 - val\_opt\_loss: 0.1142 - val\_sar\_loss: 1.1613 - val\_fusion\_loss: 0.1141 - val\_loss: 1.3896

Epoch 00008: val\_loss did not improve from 1.35987

Epoch 9/100

108/108 [=====] - 18s 165ms/step - opt\_accuracy: 0.9062 - sar\_accuracy: 0.7369 - fus\_accuracy: 0.9050 - opt\_loss: 0.0482 - sar\_loss: 1.3977 - fusion\_loss: 0.0488 - loss: 1.4948 - val\_opt\_accuracy: 0.8619 - val\_sar\_accuracy: 0.7547 - val\_fus\_accuracy: 0.8610 - val\_opt\_loss: 0.1503 - val\_sar\_loss: 1.1756 - val\_fusion\_loss: 0.1501 - val\_loss: 1.4760

Epoch 00009: val\_loss did not improve from 1.35987

Epoch 10/100

108/108 [=====] - 18s 164ms/step - opt\_accuracy: 0.9082 - sar\_accuracy: 0.7384 - fus\_accuracy: 0.9072 - opt\_loss: 0.0457 - sar\_loss: 1.3788 - fusion\_loss: 0.0463 - loss: 1.4709 - val\_opt\_accuracy: 0.8636 - val\_sar\_accuracy: 0.7559 - val\_fus\_accuracy: 0.8632 - val\_opt\_loss: 0.1669 - val\_sar\_loss: 1.2056 - val\_fusion\_loss: 0.1673 - val\_loss: 1.5398

Epoch 00010: val\_loss did not improve from 1.35987

Epoch 11/100

108/108 [=====] - 18s 164ms/step - opt\_accuracy: 0.9122 - sar\_accuracy: 0.7392 - fus\_accuracy: 0.9114 - opt\_loss: 0.0428 - sar\_loss: 1.3818 - fusion\_loss: 0.0431 - loss: 1.4677 - val\_opt\_accuracy: 0.8614 - val\_sar\_accuracy: 0.7562 - val\_fus\_accuracy: 0.8609 - val\_opt\_loss: 0.1855 - val\_sar\_loss: 1.2053 - val\_fusion\_loss: 0.1862 - val\_loss: 1.5770

Epoch 00011: val\_loss did not improve from 1.35987

Epoch 12/100

108/108 [=====] - 18s 166ms/step - opt\_accuracy: 0.9149 - sar\_accuracy: 0.7401 - fus\_accuracy: 0.9142 - opt\_loss: 0.0405 - sar\_loss: 1.3891 - fusion\_loss: 0.0409 - loss: 1.4706 - val\_opt\_accuracy: 0.8627 - val\_sar\_accuracy: 0.7571 - val\_fus\_accuracy: 0.8621 - val\_opt\_loss: 0.1408 - val\_sar\_loss: 1.1984 - val\_fusion\_loss: 0.1415 - val\_loss: 1.4807

Epoch 00012: val\_loss did not improve from 1.35987

Epoch 13/100

108/108 [=====] - 18s 167ms/step - opt\_accuracy: 0.9179 - sar\_accuracy: 0.7407 - fus\_accuracy: 0.9172 - opt\_loss: 0.0382 - sar\_loss: 1.3810 - fusion\_loss: 0.0385 - loss: 1.4577 - val\_opt\_accuracy: 0.8628 - val\_sar\_accuracy: 0.7597 - val\_fus\_accuracy: 0.8618 - val\_opt\_loss: 0.1938 - val\_sar\_loss: 1.1868 - val\_fusion\_loss: 0.1952 - val\_loss: 1.5757

Epoch 00013: val\_loss did not improve from 1.35987

Epoch 00013: early stopping

In [15]:

```
# Test Loop
time_ts = []
n_pool = 3
n_rows = 5
n_cols = 4
```

```

rows, cols = image_array.shape[:2]
pad_rows = rows - np.ceil(rows/(n_rows*2**n_pool))*n_rows*2**n_pool
pad_cols = cols - np.ceil(cols/(n_cols*2**n_pool))*n_cols*2**n_pool
print(pad_rows, pad_cols)

npad = ((0, int(abs(pad_rows))), (0, int(abs(pad_cols))), (0, 0))
image1_pad = np.pad(image_array, pad_width=npad, mode='reflect')

h, w, c = image1_pad.shape
patch_size_rows = h//n_rows
patch_size_cols = w//n_cols
num_patches_x = int(h/patch_size_rows)
num_patches_y = int(w/patch_size_cols)

input_shape=(patch_size_rows,patch_size_cols, c)

#if method == 'unet':
#    new_model = build_unet(input_shape, nb_filters, number_class)

#if method == 'resunet':
#    new_model = build_resunet(input_shape, nb_filters, number_class)

new_model = Model_3(nb_filters, number_class, n_opt_layers)
new_model.build((None,)+input_shape)
adam = Adam(lr = 1e-3 , beta_1=0.9)
loss = weighted_categorical_crossentropy(weights)
new_model.compile(optimizer=adam, loss=loss, metrics=['accuracy'], run_eagerly=True)

for tm in range(0,times):
    print('time: ', tm)
    #model = Load_model(path_models+ '/' + method + '_' +str(tm)+'.h5', compile=False)

    #for l in range(1, len(model.layers)):
    #    new_model.layers[l].set_weights(model.layers[l].get_weights())
    new_model.load_weights(path_models+ '/' + method + '_' +str(tm)+'.h5')

    start_test = time.time()
    patch_opt = []
    patch_sar = []
    patch_fus = []
    patch_comb = []

    for i in range(0,num_patches_y):
        for j in range(0,num_patches_x):
            patch = image1_pad[patch_size_rows*j:patch_size_rows*(j+1), patch_size_c
            pred_opt, pred_sar, pred_fus, pred_comb = new_model.predict(np.expand_di
            del patch
            patch_opt.append(pred_opt[:, :, :, 1])
            patch_sar.append(pred_sar[:, :, :, 1])
            patch_fus.append(pred_fus[:, :, :, 1])
            patch_comb.append(pred_comb[:, :, :, 1])
            del pred_opt, pred_sar, pred_fus, pred_comb
    end_test = time.time() - start_test

    patches_pred_opt = np.asarray(patch_opt).astype(np.float32)
    patches_pred_sar = np.asarray(patch_sar).astype(np.float32)
    patches_pred_fus = np.asarray(patch_fus).astype(np.float32)
    patches_pred_comb = np.asarray(patch_comb).astype(np.float32)

    prob_reconstructed_opt = pred_reconstruct(h, w, num_patches_x, num_patches_y, pat
    prob_reconstructed_sar = pred_reconstruct(h, w, num_patches_x, num_patches_y, pat
    prob_reconstructed_fus = pred_reconstruct(h, w, num_patches_x, num_patches_y, pat
    prob_reconstructed_comb = pred_reconstruct(h, w, num_patches_x, num_patches_y, pa

    del patches_pred_opt, patches_pred_sar, patches_pred_fus, patches_pred_comb

```

```

np.save(path_maps+'/'+ 'prob_opt_'+str(tm)+'.npy', prob_reconstructed_opt)
np.save(path_maps+'/'+ 'prob_sar_'+str(tm)+'.npy', prob_reconstructed_sar)
np.save(path_maps+'/'+ 'prob_fus_'+str(tm)+'.npy', prob_reconstructed_fus)
np.save(path_maps+'/'+ 'prob_comb_'+str(tm)+'.npy', prob_reconstructed_comb)

time_ts.append(end_test)
del prob_reconstructed_opt, prob_reconstructed_sar, prob_reconstructed_fus, prob_re
#del model
time_ts_array = np.asarray(time_ts)
# Save test time
np.save(path_exp+'/metrics_ts.npy', time_ts_array)

0.0 -8.0
time: 0
time: 1
time: 2
time: 3
time: 4

```

In [16]:

```

# Compute mean of the tm predictions maps
prob_rec_opt = np.zeros((image1_pad.shape[0], image1_pad.shape[1], times))
prob_rec_sar = np.zeros((image1_pad.shape[0], image1_pad.shape[1], times))
prob_rec_fus = np.zeros((image1_pad.shape[0], image1_pad.shape[1], times))
prob_rec_comb = np.zeros((image1_pad.shape[0], image1_pad.shape[1], times))

for tm in range(0, times):
    print(tm)
    prob_rec_opt[:, :, tm] = np.load(path_maps+'/'+ 'prob_opt_'+str(tm)+'.npy').astype(
    prob_rec_sar[:, :, tm] = np.load(path_maps+'/'+ 'prob_sar_'+str(tm)+'.npy').astype(
    prob_rec_fus[:, :, tm] = np.load(path_maps+'/'+ 'prob_fus_'+str(tm)+'.npy').astype(
    prob_rec_comb[:, :, tm] = np.load(path_maps+'/'+ 'prob_comb_'+str(tm)+'.npy').astyp

mean_prob_opt = np.mean(prob_rec_opt, axis = -1)
mean_prob_sar = np.mean(prob_rec_sar, axis = -1)
mean_prob_fus = np.mean(prob_rec_fus, axis = -1)
mean_prob_comb = np.mean(prob_rec_comb, axis = -1)

np.save(path_maps+'/prob_mean_opt.npy', mean_prob_opt)
np.save(path_maps+'/prob_mean_sar.npy', mean_prob_sar)
np.save(path_maps+'/prob_mean_fus.npy', mean_prob_fus)
np.save(path_maps+'/prob_mean_comb.npy', mean_prob_comb)

```

```

0
1
2
3
4

```

In [17]:

```

# Plot mean map and reference
fig = plt.figure(figsize=(20,10))
ax1 = fig.add_subplot(151)
plt.title('OPT Prediction')
ax1.imshow(mean_prob_opt, cmap = 'jet')
ax1.axis('off')

ax1 = fig.add_subplot(152)
plt.title('SAR Prediction')
ax1.imshow(mean_prob_sar, cmap = 'jet')
ax1.axis('off')

ax1 = fig.add_subplot(153)
plt.title('FUSION Prediction')
ax1.imshow(mean_prob_fus, cmap = 'jet')
ax1.axis('off')

```



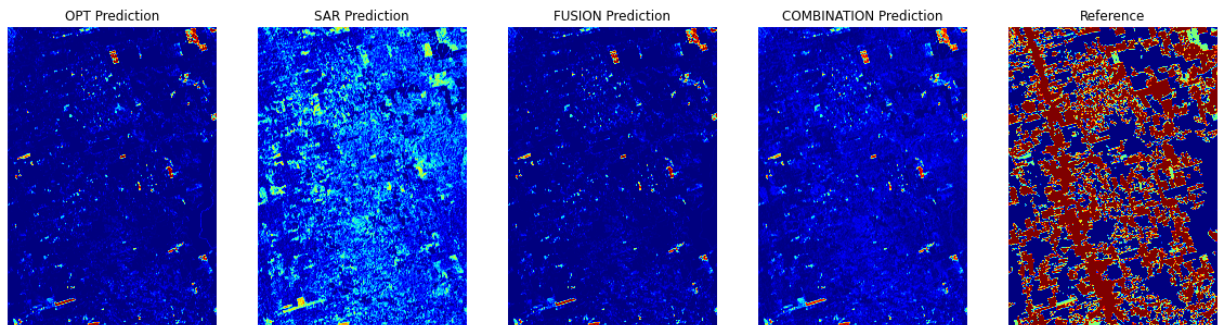
```

ax1 = fig.add_subplot(154)
plt.title('COMBINATION Prediction')
ax1.imshow(mean_prob_comb, cmap = 'jet')
ax1.axis('off')

ax2 = fig.add_subplot(155)
plt.title('Reference')
ax2.imshow(final_mask1, cmap = 'jet')
ax2.axis('off')

```

Out[17]: (-0.5, 6999.5, 9999.5, -0.5)



In [18]:

```

# Computing metrics
mean_prob_opt = mean_prob_opt[:final_mask1.shape[0], :final_mask1.shape[1]]
mean_prob_sar = mean_prob_sar[:final_mask1.shape[0], :final_mask1.shape[1]]
mean_prob_fus = mean_prob_fus[:final_mask1.shape[0], :final_mask1.shape[1]]
mean_prob_comb = mean_prob_comb[:final_mask1.shape[0], :final_mask1.shape[1]]

ref1 = np.ones_like(final_mask1).astype(np.float32)

ref1[final_mask1 == 2] = 0
TileMask = mask_amazon_ts * ref1
GTTruePositives = final_mask1==1

Npoints = 10

Pmax_opt = np.max(mean_prob_opt[GTTruePositives * TileMask ==1])
ProbList_opt = np.linspace(Pmax_opt,0,Npoints)

Pmax_sar = np.max(mean_prob_sar[GTTruePositives * TileMask ==1])
ProbList_sar = np.linspace(Pmax_sar,0,Npoints)

Pmax_fus = np.max(mean_prob_fus[GTTruePositives * TileMask ==1])
ProbList_fus = np.linspace(Pmax_fus,0,Npoints)

Pmax_comb = np.max(mean_prob_comb[GTTruePositives * TileMask ==1])
ProbList_comb = np.linspace(Pmax_comb,0,Npoints)

metrics_opt = matrices_AA_recall(ProbList_opt, mean_prob_opt, final_mask1, mask_amazon_ts)
metrics_sar = matrices_AA_recall(ProbList_sar, mean_prob_sar, final_mask1, mask_amazon_ts)
metrics_fus = matrices_AA_recall(ProbList_fus, mean_prob_fus, final_mask1, mask_amazon_ts)
metrics_comb = matrices_AA_recall(ProbList_comb, mean_prob_comb, final_mask1, mask_amazon_ts)

np.save(path_exp+'acc_metrics_opt.npy',metrics_opt)
np.save(path_exp+'acc_metrics_sar.npy',metrics_sar)
np.save(path_exp+'acc_metrics_fus.npy',metrics_fus)
np.save(path_exp+'acc_metrics_comb.npy',metrics_comb)

```

0.9983846426010132

D:\Ferrari\proj\_1\projeto\utils\_unet\_resunet.py:200: RuntimeWarning: invalid value encountered in longlong\_scalars

```

precision_ = TP/(TP+FP)
0.8874530156453451
0.7765213886896769
0.6655897617340089
0.5546581347783407
0.44372650782267253
0.3327948808670045
0.22186325391133632
0.11093162695566816
0.0
0.5149446249008178
D:\Ferrari\proj_1\projeto\utils_unet_resunet.py:200: RuntimeWarning: invalid value encountered in longlong_scalars
precision_ = TP/(TP+FP)
0.4577285554673936
0.40051248603396944
0.3432964166005452
0.28608034716712105
0.22886427773369683
0.1716482083002726
0.11443213886684844
0.05721606943342422
0.0
0.9972945213317871
D:\Ferrari\proj_1\projeto\utils_unet_resunet.py:200: RuntimeWarning: invalid value encountered in longlong_scalars
precision_ = TP/(TP+FP)
0.8864840189615886
0.7756735165913899
0.6648630142211913
0.5540525118509928
0.4432420094807942
0.33243150711059566
0.22162100474039714
0.11081050237019852
0.0
0.8240293860435486
D:\Ferrari\proj_1\projeto\utils_unet_resunet.py:200: RuntimeWarning: invalid value encountered in longlong_scalars
precision_ = TP/(TP+FP)
0.7324705653720431
0.6409117447005378
0.5493529240290325
0.457794103357527
0.3662352826860216
0.27467646201451623
0.1831176413430109
0.09155882067150545
0.0

```

In [19]:

```

# Complete NaN values
metrics_copy_opt = metrics_opt.copy()
metrics_copy_opt = complete_nan_values(metrics_copy_opt)

metrics_copy_sar = metrics_sar.copy()
metrics_copy_sar = complete_nan_values(metrics_copy_sar)

metrics_copy_fus = metrics_fus.copy()
metrics_copy_fus = complete_nan_values(metrics_copy_fus)

metrics_copy_comb = metrics_comb.copy()
metrics_copy_comb = complete_nan_values(metrics_copy_comb)

```

```

In [20]: # Compute Mean Average Precision (mAP) score
Recall_opt = metrics_copy_opt[:,0]
Precision_opt = metrics_copy_opt[:,1]
AA_opt = metrics_copy_opt[:,2]

Recall_sar = metrics_copy_sar[:,0]
Precision_sar = metrics_copy_sar[:,1]
AA_sar = metrics_copy_sar[:,2]

Recall_fus = metrics_copy_fus[:,0]
Precision_fus = metrics_copy_fus[:,1]
AA_fus = metrics_copy_fus[:,2]

Recall_comb = metrics_copy_comb[:,0]
Precision_comb = metrics_copy_comb[:,1]
AA_comb = metrics_copy_comb[:,2]

DeltaR_opt = Recall_opt[1:]-Recall_opt[:-1]
AP_opt = np.sum(Precision_opt[:-1]*DeltaR_opt)
print('OPT mAP', AP_opt)

DeltaR_sar = Recall_sar[1:]-Recall_sar[:-1]
AP_sar = np.sum(Precision_sar[:-1]*DeltaR_sar)
print('SAR mAP', AP_sar)

DeltaR_fus = Recall_fus[1:]-Recall_fus[:-1]
AP_fus = np.sum(Precision_fus[:-1]*DeltaR_fus)
print('FUSION mAP', AP_fus)

DeltaR_comb = Recall_comb[1:]-Recall_comb[:-1]
AP_comb = np.sum(Precision_comb[:-1]*DeltaR_comb)
print('COMBINATION mAP', AP_comb)

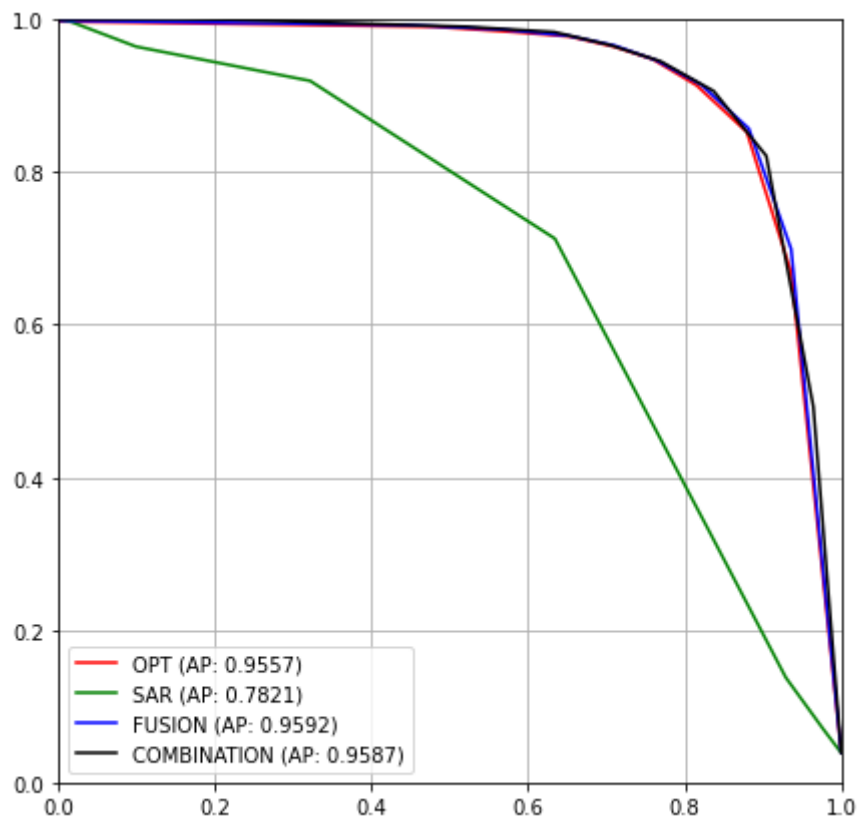
# Plot Recall vs. Precision curve
plt.figure(figsize=(7,7))
plt.plot(metrics_copy_opt[:,0],metrics_copy_opt[:,1], 'r-', label = f'OPT (AP: {AP_o
plt.plot(metrics_copy_sar[:,0],metrics_copy_sar[:,1], 'g-', label = f'SAR (AP: {AP_s
plt.plot(metrics_copy_fus[:,0],metrics_copy_fus[:,1], 'b-', label = f'FUSION (AP: {A
plt.plot(metrics_copy_comb[:,0],metrics_copy_comb[:,1], 'k-', label = f'COMBINATION
plt.legend(loc="lower left")
ax = plt.gca()
ax.set_ylim([0,1])
ax.set_xlim([0,1])
#plt.plot(metrics_copy[:,0],metrics_copy[:,2])
plt.grid()

```

```

OPT mAP 0.9556519922943457
SAR mAP 0.7820542521444049
FUSION mAP 0.959235171891128
COMBINATION mAP 0.9586848670660806

```



In [ ]: