BENCHMARK SCNN+Attention

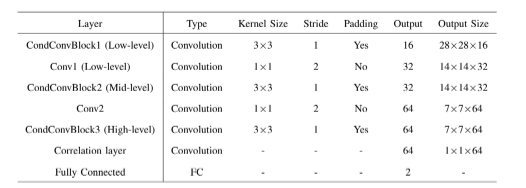
In this document, we are utilizing SAFNet code for Ottawa and Yellow River dataset. For training the model, we are utilizing the below parameters:

|  |  |
| --- | --- |
| Parameters | Value |
| Train ratio | 0.9 |
| Test ratio | 0.2 |
| Epochs | 50 |
| Batch Size | 128 |
| Learning Rate | 0.001 |
| Momentum | 0.9 |
| Betas | (0.9,0.999) |

Training Method for SAFNet(Benchmark):

Given images I1 and I2, we first create image patches with window size of 7\*7, such that the ground truth value corresponding to those patches is greater than 0. Then, these patches are fed into the CNN for feature extraction. By feature extraction using CNN, the images are less sensitive to speckle noise.

These images are then fed into Siamese adaptive fusion network followed by the proposed SAFNet, whose structure is given below:



Training Method for SCNN:

First image patches of window size 7\*7 is created as done before for SAFNet. Then, the SCNNModel is trained by feeding two SAR images through convolutional layers, merging their outputs with pairwise connections, and passing them through fully connected layers. The model learns to predict the segmentation of each image by adjusting its parameters to minimize the difference between predicted and actual segmentations during training.

Activation function: Leaky ReLU

Method for down-sampling: MaxPool2D

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Layer | Kernel Size | Padding | Stride | Output |
| Convolutional | 3\*3 | 1 | 2 | 64 |
| Convolutional | 3\*3 | 1 | 2 | 128 |
| Convolutional | 3\*3 | 1 | 2 | 256 |
| Convolutional | 3\*3 | 1 | 2 | 256 |
| Convolutional | 3\*3 | 1 | 2 | 256 |
| Convolutional (SAR Output 1) | 3\*3 | 1 | 2 | 256 |
| Convolutional (SAR Output 2) | 3\*3 | 1 | 2 | 256 |
| Convolutional (Pair-Wise Skipped Connection) | 1\*1 | 0 |  | 256 |
| Convolutional (Pair-Wise Skipped Connection) | 1\*1 | 0 |  | 256 |

Training Method for SCNN+Attention:

The same method is applied as given above, but we have utilized attention mechanisms after convolution blocks with input channel as 512 and output channel as 1 with kernel of 1\*1 size. This is done to reduce the channel dimension from 512 to 1.

Yellow River dataset:

Image shape=291\*306

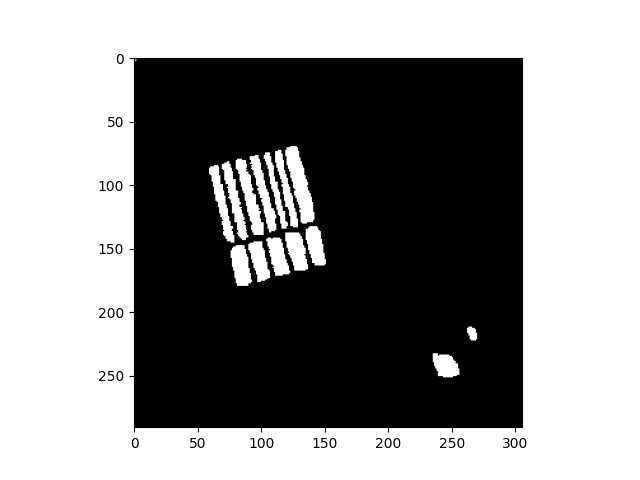


Fig:Ground Truth Image

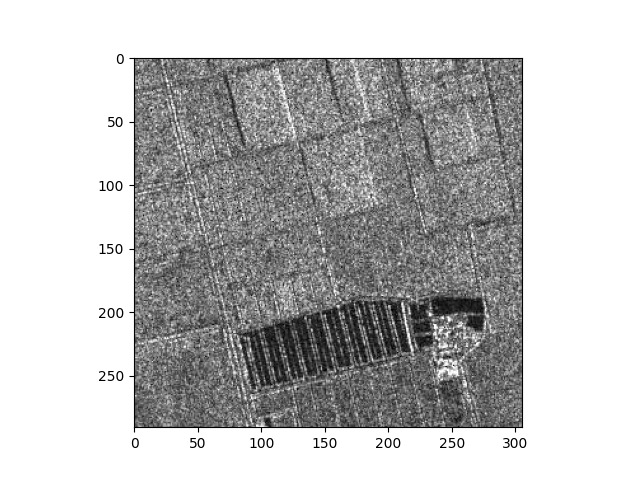


Fig: Image 1 of yellow river dataset

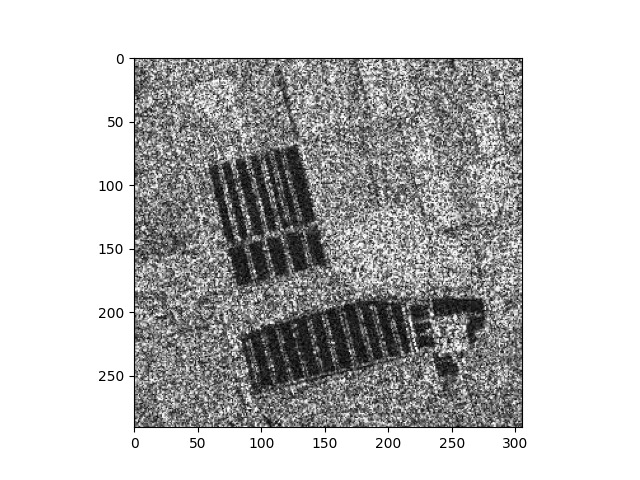


Fig: Image 2 for yellow river dataset

RESULT for Yellow River dataset:

|  |  |  |  |
| --- | --- | --- | --- |
|  | SAFNet | SCNN | SCNN+Attention |
| Training Loss | 0.05 | 0.00 | 0.00 |
| Training accuracy | 99.78% | 100% | 100% |
| Testing accuracy | 94.37% | 95.75% | 95.75% |

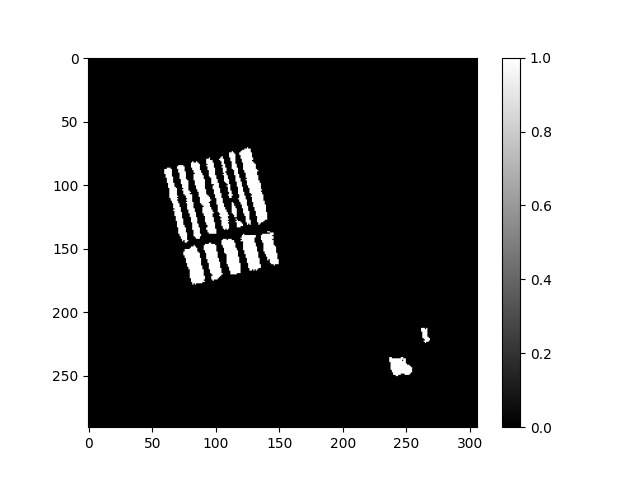


Fig:Predicted change map for yellow river using SAFNet

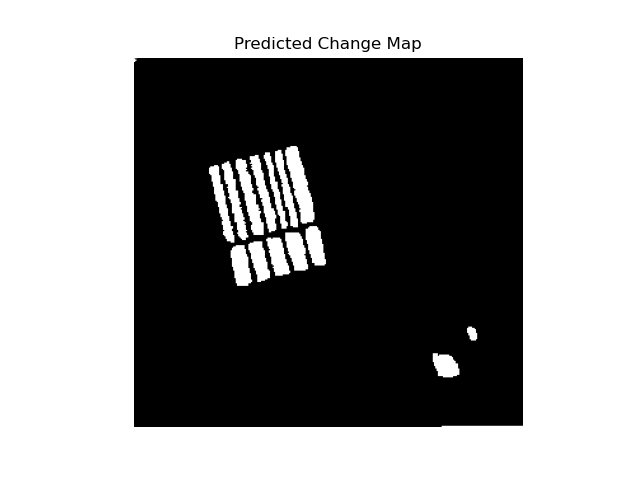


Fig: Predicted Change Map for yellow river using SCNN

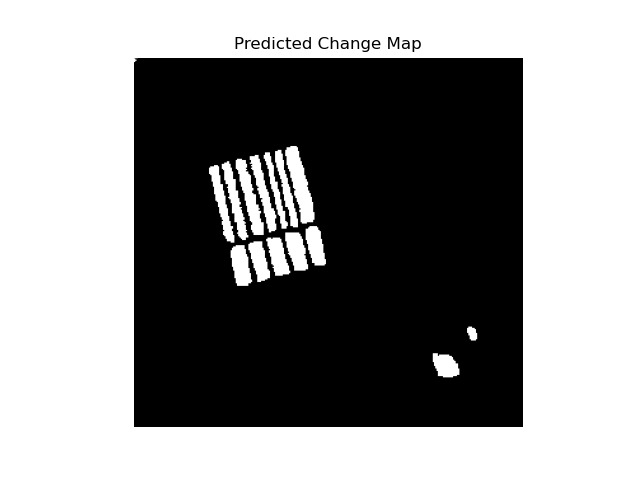


Fig: Predicted Change Map for yellow river using SCNN+Attention

Ottawa Dataset:

Shape: 350\*290

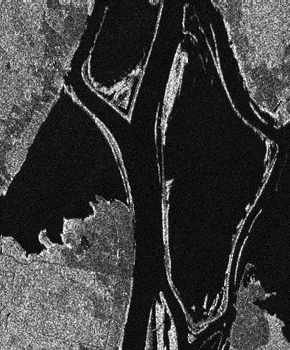


Fig 1: Image 1 for Ottawa dataset

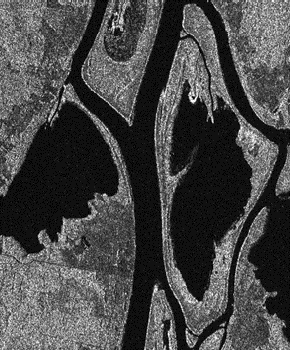


Fig 2: Image 2 for Ottawa dataset

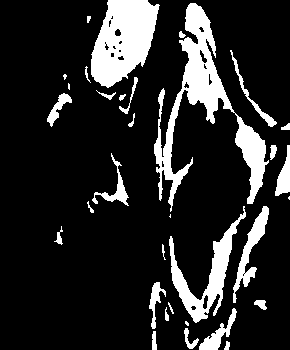


Fig 3: Ground truth image

RESULT for Ottawa dataset:

|  |  |  |  |
| --- | --- | --- | --- |
|  | SAFNet | SCNN | SCNN+Attention |
| Training Loss | 0.0002 | 0.0005 | 0.00 |
| Training accuracy | 100% | 100% | 100% |
| Testing accuracy | 100% | 100% | 100% |

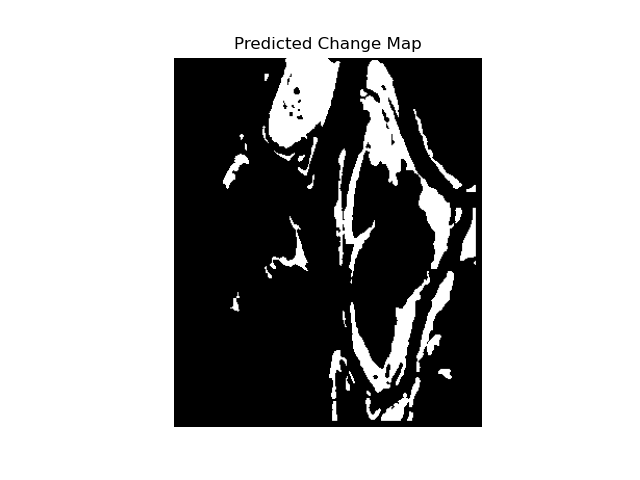


Fig: Predicted Change Map for Ottawa using SAFNet

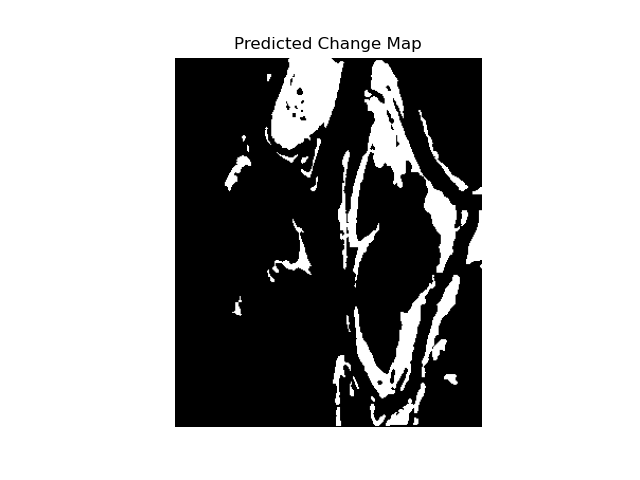


Fig: Predicted Change Map for Ottawa using SCNN

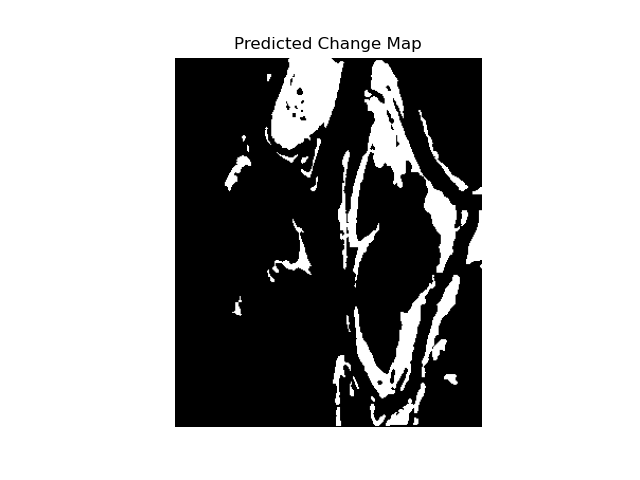


Fig: Predicted Change Map for Ottawa using SCNN+Attention