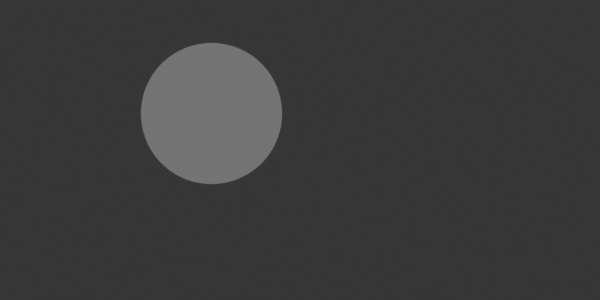
**Change Detection of SAR Images Through the Use of RBM Results**

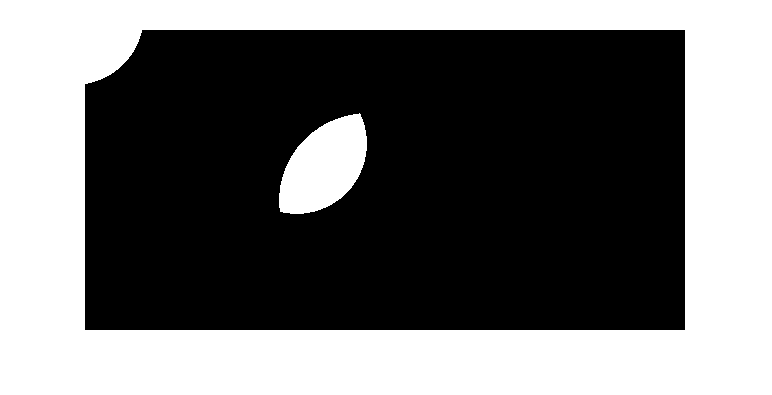
**Experiment 1:**

**Hypothesis:** The network will perform worse the higher the noise variance in artificially changed images with speckle noise.

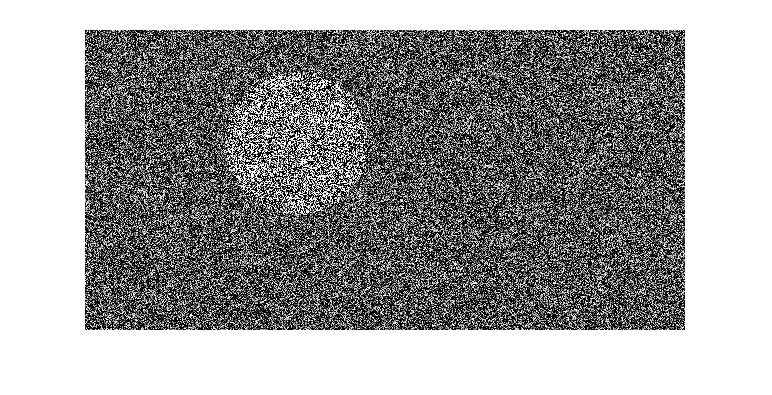
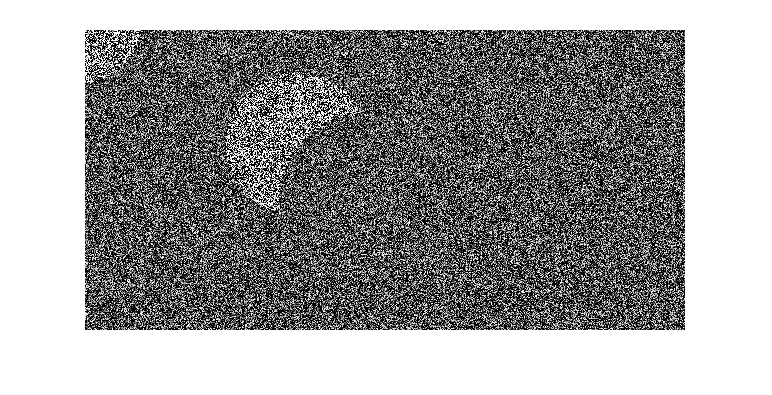
**Method:**

**Step 1:** Create Artificial Images to Test With:

**Step 2:** Create Ground Truth Map



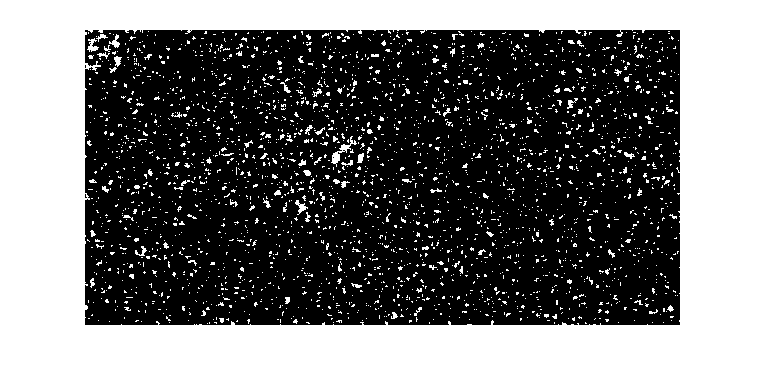
**Step 3:** Create different levels of speckle noise on the artificial images using imnoise() function in Matlab, controlled by changing the variance of noise.

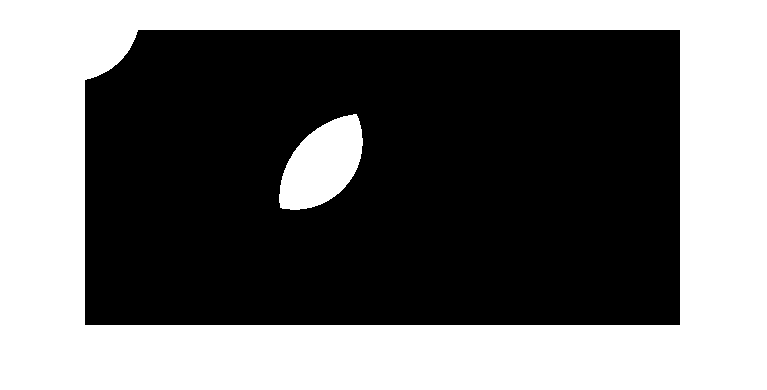
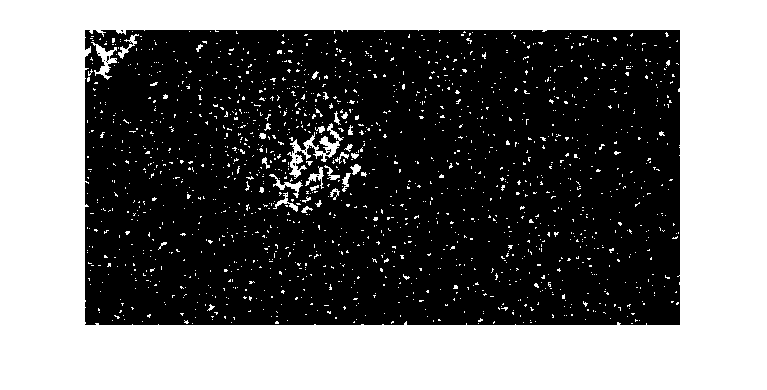
 Above image with variance 5:

**Step 4:** Run images through network and compare the output to the ground truth. The measurement of accuracy is percentage correct classification (PCC) and is determined by the following equation:

Where TP are the true positives (pixels classified correctly as changed), TN are the true negatives (pixels classified correctly as unchanged), FP are the false positives (pixels incorrectly classified as changed), and FN are the false negatives (pixels incorrectly classified as unchanged).

**Results:**

Sample Results, with variance 0, 5, and 10, respectively:

99.5% PCC 93.6% PCC 89.9% PCC

Continuation of Set 8:

**Conclusion:**

****The network’s performance decreases when the amount of noise increases. Set 8 was an outlier because the network currently performs exceptionally well the darker the contrast, if the change is from dark to light. Eventually, adding enough variance broke the network for set 8 as well. Here is the image set from set 8, for reference:

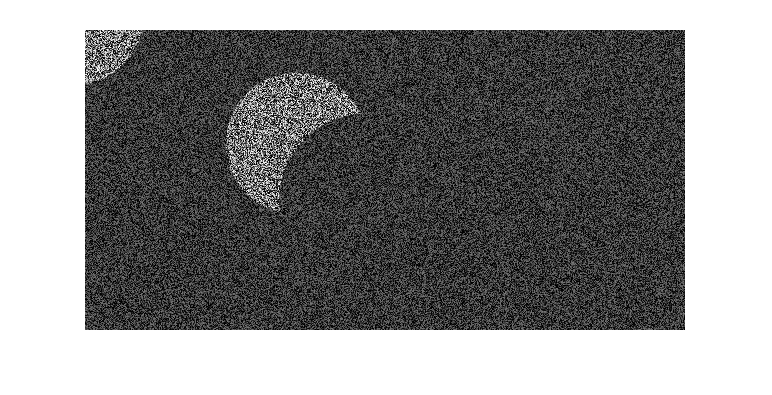
****

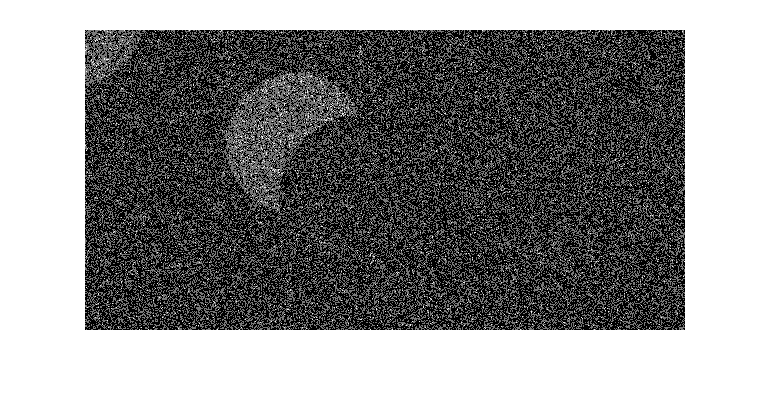
**Experiment 2:**

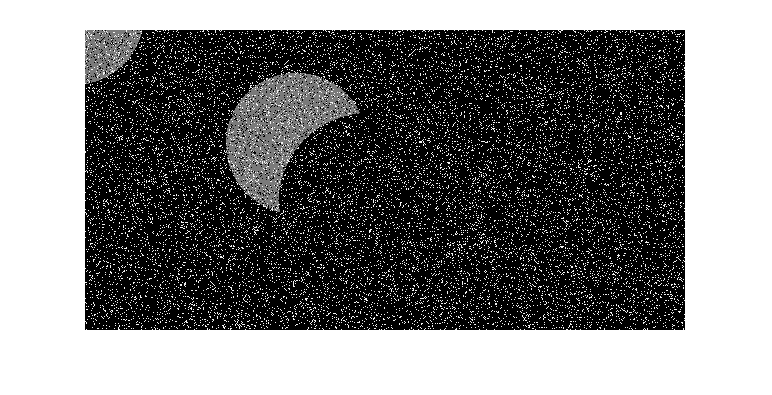
**Hypothesis:** The network performs better with different types of noise.

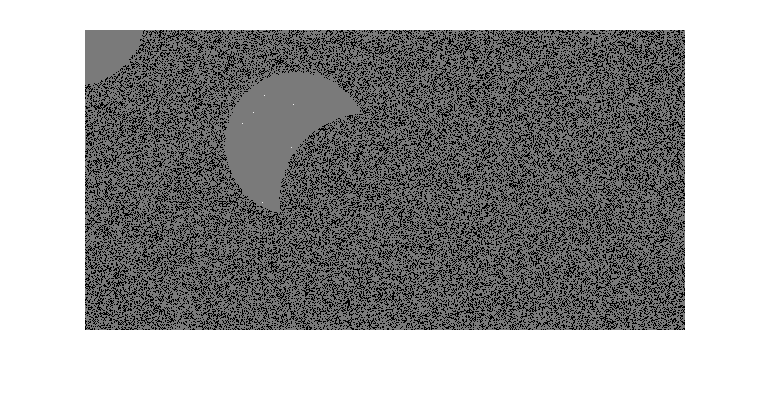
**Method:**

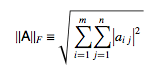
**Step 1:** The following noises were added to the test images:

* **Speckle noise:** Uniformly distributed random noise with a mean of 0 and a variance of .5, a type of multiplicative noise:
* **Gaussian noise:** Gaussian white noise with a mean of 0 and a variance of .1, a type of additive noise:



* **Salt & Pepper Noise:** On and off pixels, with a noise density of .2, a type of additive noise:
* **Poisson Noise:** A type of noise that mimics electronic noise, also a type of additive noise:



**Step 2:** The images were normalized through the following steps:

* Find the Frobenious norm for each image through the following equation:
* Divide each pixel by the Frobenious norm of the corresponding image and multiply by the Frobenious norm of the speckle image

**Step 3:** Run images through network and compare the output to the ground truth.

**Step 4:** Repeat experiment 1 with Gaussian noise, speckle noise, and salt & pepper noise, without normalizing the images this time to observe how the network responds to each noise independently.

**Results:**

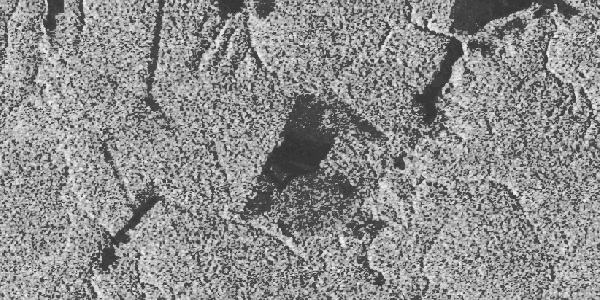
**Conclusion:**

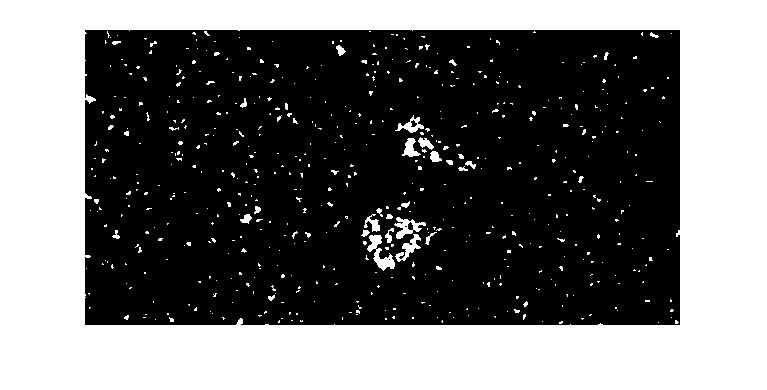
The results are inconclusive. It is difficult to find a comparable amount of noise for each of these types to test fully, and the bottom and upper threshold for changed pixels in the network changes depending on the image and the noise. Without the optimal thresholds for each type of noise, it is difficult to compare the different types. However, based on the results, the network can interpret images with Gaussian noise, speckle noise, and salt and pepper noise. As well as this, regardless of noise type, the higher the variance and density of the noise, the worse the network will perform.

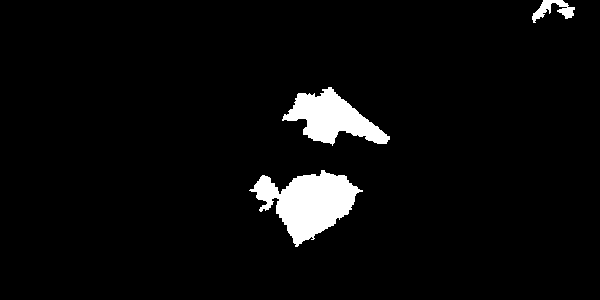
Finally, here are results for actual SAR images, as well as another artificial image:

|  |  |
| --- | --- |
| Set # | PCC |
| 1 | 97.9% |
| 2 | 94.6% |
| 3 | 98.8% |
| 4 | 99.8% |

1:



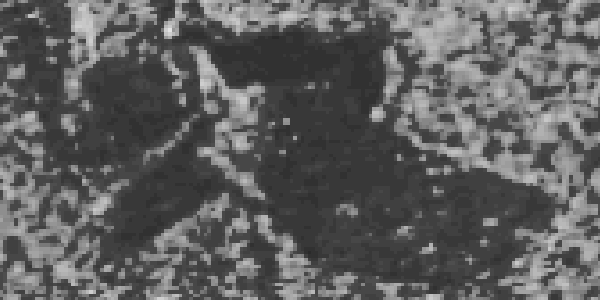
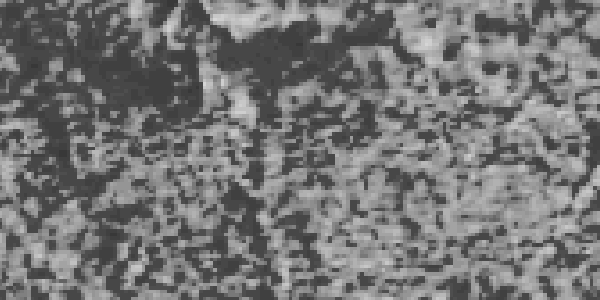


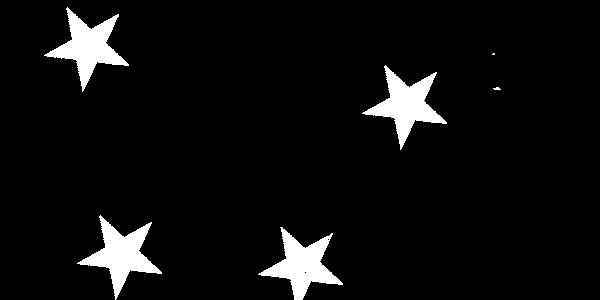


2:

3:





4: