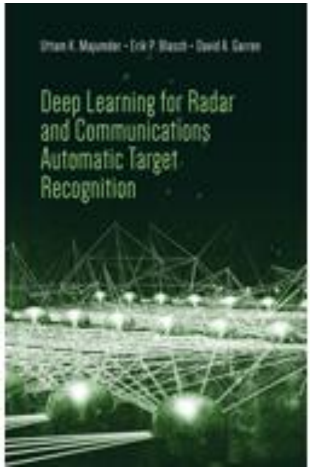
Machine Learning Techniques for Radar Automatic Target Recognition (ATR)

Chapter 7:
Deep Learning for Multiple Targets Classification

Uttam K. Majumder

Machine Learning Techniques for Radar Automatic Target Recognition (ATR)

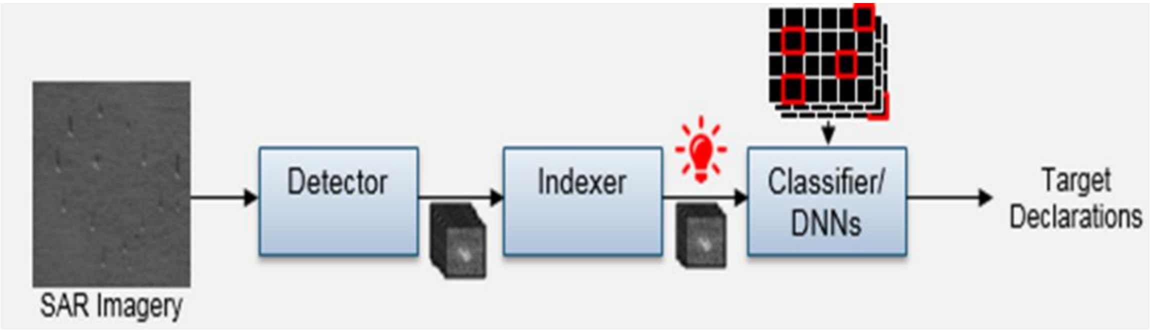
Lecture Outline



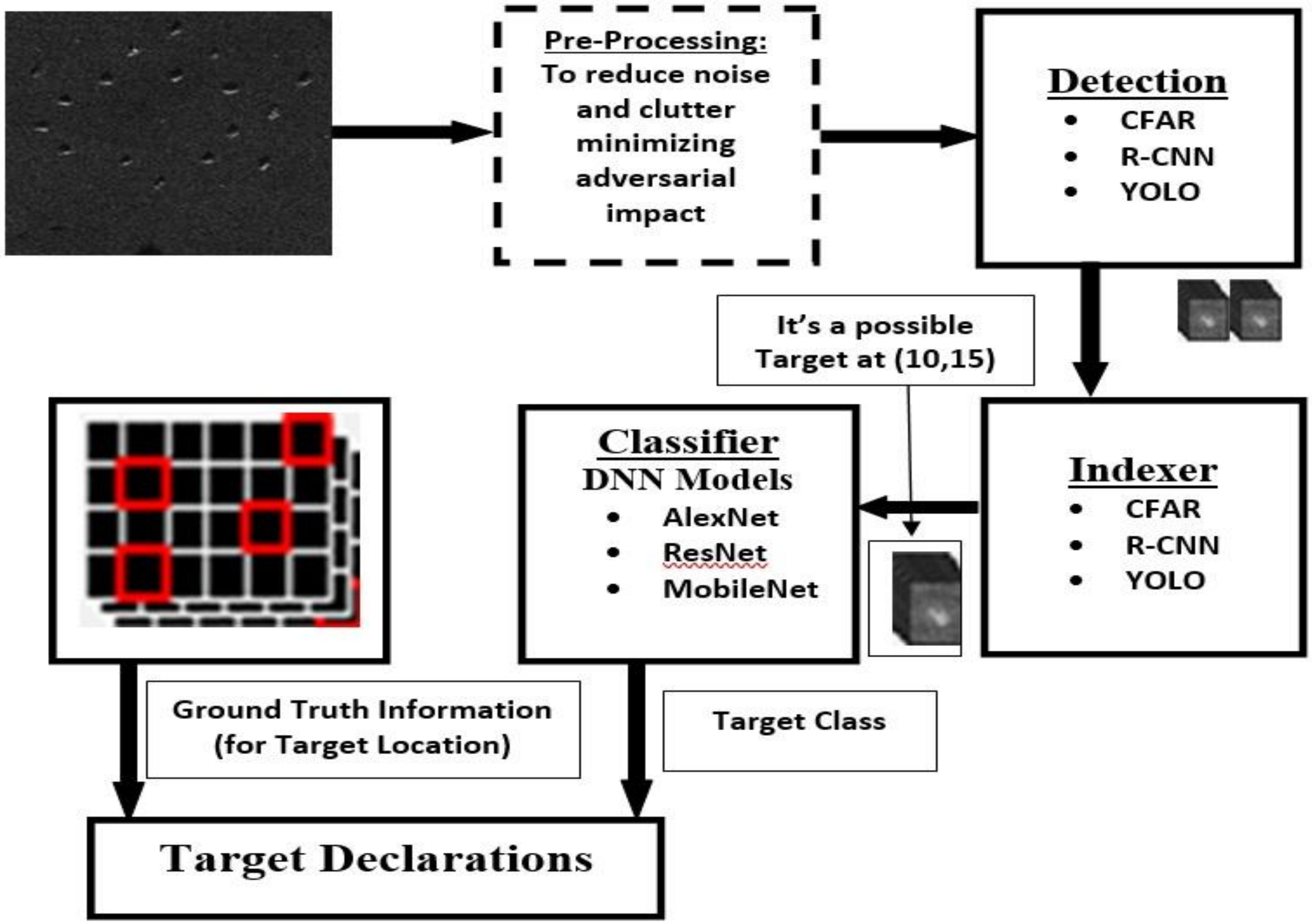
1. **Radio Frequency ATR: Past, Present, and Future: 20 min**
2. **Mathematics for Machine Learning / Deep Learning: 20 min**
3. Review of ML Algorithms: 25 min
4. **Deep Learning Algorithms: 30 min**
5. **RF Data for ML Research: 15 min**
6. **DL for Single Target Classification: 25 min**
7. **DL for Many Targets Classification: 20 min**
8. **RF Signals Classification: 20 min**
9. **RF ATR Performance Evaluation: 25 min**
10. **Emerging ML Algorithms for RF ATR: 35 min**

3.2 Multiple RF Objects Classification

- Input
  - Full Frame multi-object Synthetic Aperture Radar (SAR) Image with clutter
  - Meta data file containing the ground truth (GT) information for the image
- Process
  - Detection
    - Preprocess input image with wavelet transforms
    - Run detector on processed image and produce one 128x128 pixel jpeg image chip for each detected object
  - Classification
    - Classify each chip using a custom DCNN
- Output
  - Compare results of classification to the GT info to produce statistics and to show a visual representation of the results

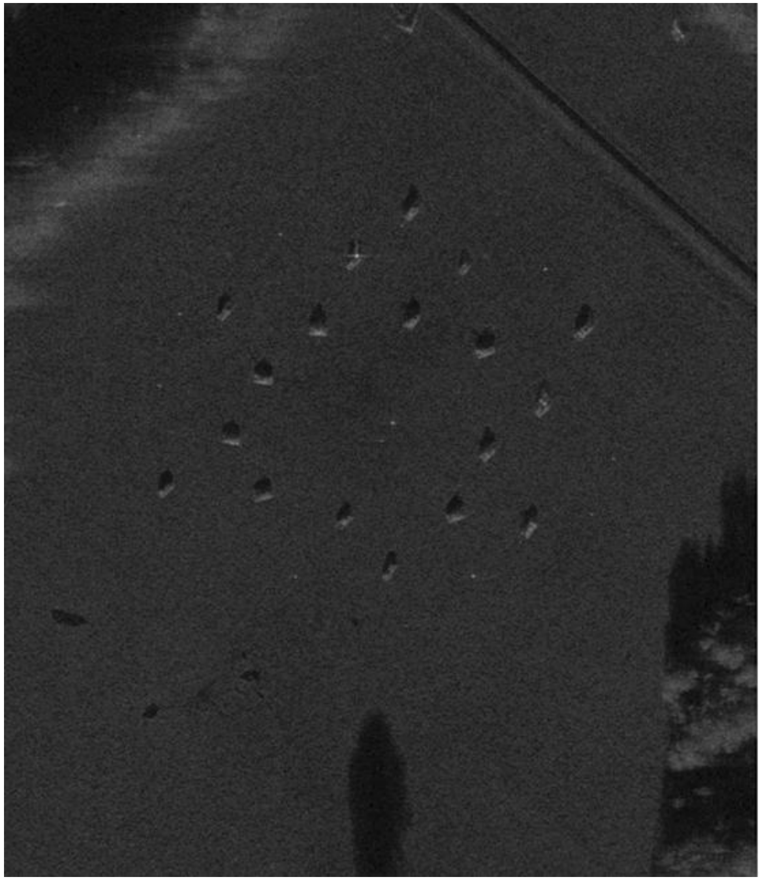


SAR Multiple-Targets Detection and Classification: Algorithmic Steps



3.2 Input Data

- **Full Frame multi-object SAR image scene**
  - **1476x1784 pixel JPG (grayscale)**
  - **Notice, the tree lines and other clutter including corner reflectors**



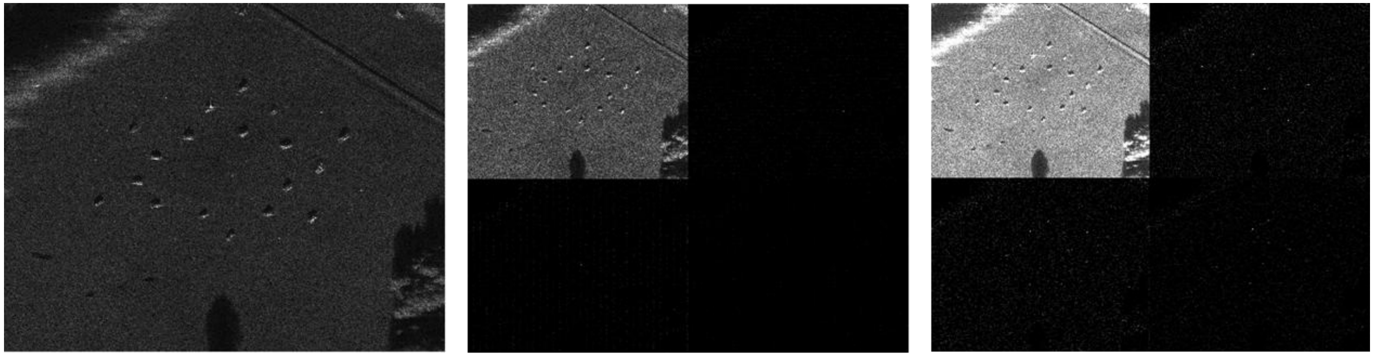
**Brightened for visual representation

<u>**XML File Sample**</u>


```
…
<DetailObjectInfo>
   <NumberOfObjectsInImage>18</NumberOfObjectsInImage>
   <NumberOfObjectsInScene>18</NumberOfObjectsInScene>
   <Object>
     <SystemName>BTR70</SystemName>
     <Function>Armored Personnel Carrier</Function>
     <SerialNumber>C71</SerialNumber>
     <Orientation>
       <Roll>-1.179039 deg</Roll>
       <Pitch>358.580475 deg</Pitch>
       <Yaw>351.30838 deg</Yaw>
     </Orientation>
     <AzimuthAngle>-28.006783 deg</AzimuthAngle>
     <ImageLocation>
       <CenterPixel>
         <Row>1162</Row>
         <Col>757</Col>
       </CenterPixel>
     </ImageLocation>
     <SeasonalCover>only growing vegetation</SeasonalCover>
   </Object>
          …
</DetailObjectInfo>
…
```

3.2 2D-DWT for SAR Imagery

- **Wavelets are intriguing for use with SAR imagery due to the high-frequency speckle noise that is characteristic of this type of imagery.**
- **Expect that the objects would exist in the lower frequencies of the image, so their signatures would not get filtered out by the high pass filters of the first few scales.**
- **We use the scale 1 and scale 2 approximations (i.e. LL sub-images).**
  - **At scale 1, the approximation contains frequencies (in radians) 0 → pi/2**
  - **At scale 2, the approximation contains frequencies 0 → pi/4**
- **Since the objects in the images are relatively small, we use a two dimensional *stationary* wavelet transform (2D-SWT) which does not subsample at each scale.**
  - **Maintain the dimensionality of the original image.**

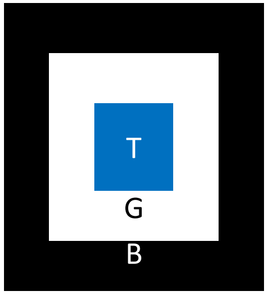


(a) (b) (c)

In the figure, (a) is the original image, (b) is the four sub-images at scale 1, and (c) is the four sub-images at scale 2.

3.2 Constant False Alarm Rate Detector (CFAR)

- **Designed to maintain a constant probability of false alarm (PFA)**
  - **Use an alpha (α) parameter to set PFA**
- **Shifts a *NxN* window across the image**
- **At each location *(x,y)*, the average of the pixels in the target region (μ_T) and the average of the pixels in the background region (μ_B) are computed and compared**
  - **Note, the pixels in the guard region are ignored**
- **CFAR Equation**
  - **Inputs:**
    - $(\boldsymbol{B}, \boldsymbol{G}, \boldsymbol{T}) = (\mathbf{60}, \mathbf{30}, \mathbf{10})$
    - $\boldsymbol{\alpha} = \mathbf{0.6}$
  - $\boldsymbol{result}(\boldsymbol{x}, \boldsymbol{y}) = \begin{cases} \mathbf{1}, \boldsymbol{\mu_T} * \boldsymbol{\alpha} > \boldsymbol{\mu_B} \\ \mathbf{0}, \boldsymbol{otherwise} \end{cases}$
  - **Output**
    - **result = binary Image**



T = Target Region
G = Guard Region
B = Background Region

3.2 2D-CFAR Results

CFAR(Original)

CFAR(Scale 1 LL)

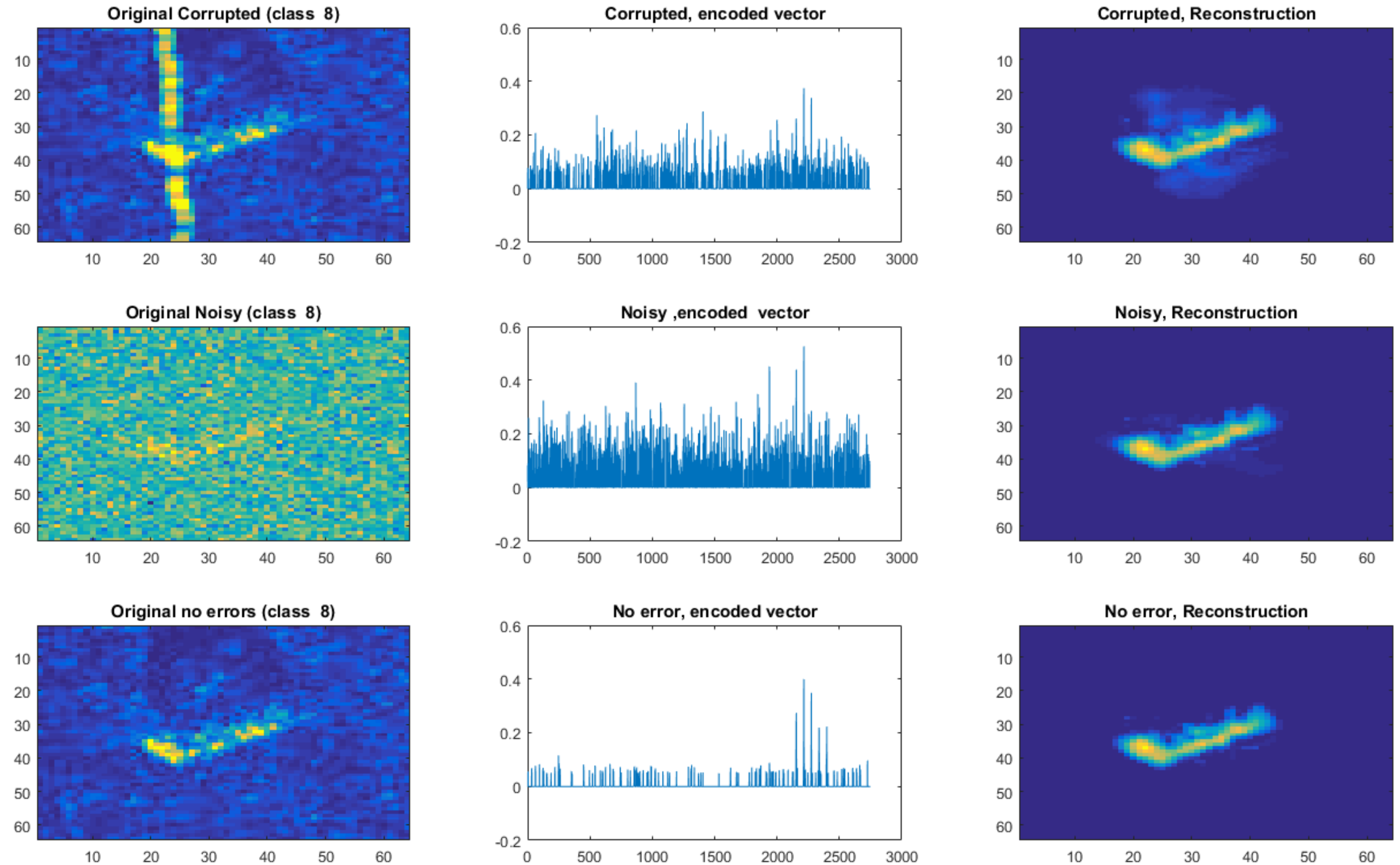
CFAR(Scale 2 LL)

Notice that the CFAR(Scale 4 LL) looks the best from this view. However, areas of white appear along the left edge of the image and the objects are slightly shifted from their original positions. It would be worth exploring this more.
(Same CFAR parameters used in all cases)

CFAR(Scale 3 LL)

CFAR(Scale 4 LL)

SAR Image Pre-Processing(ℓ_1 -min) for Noise Reduction



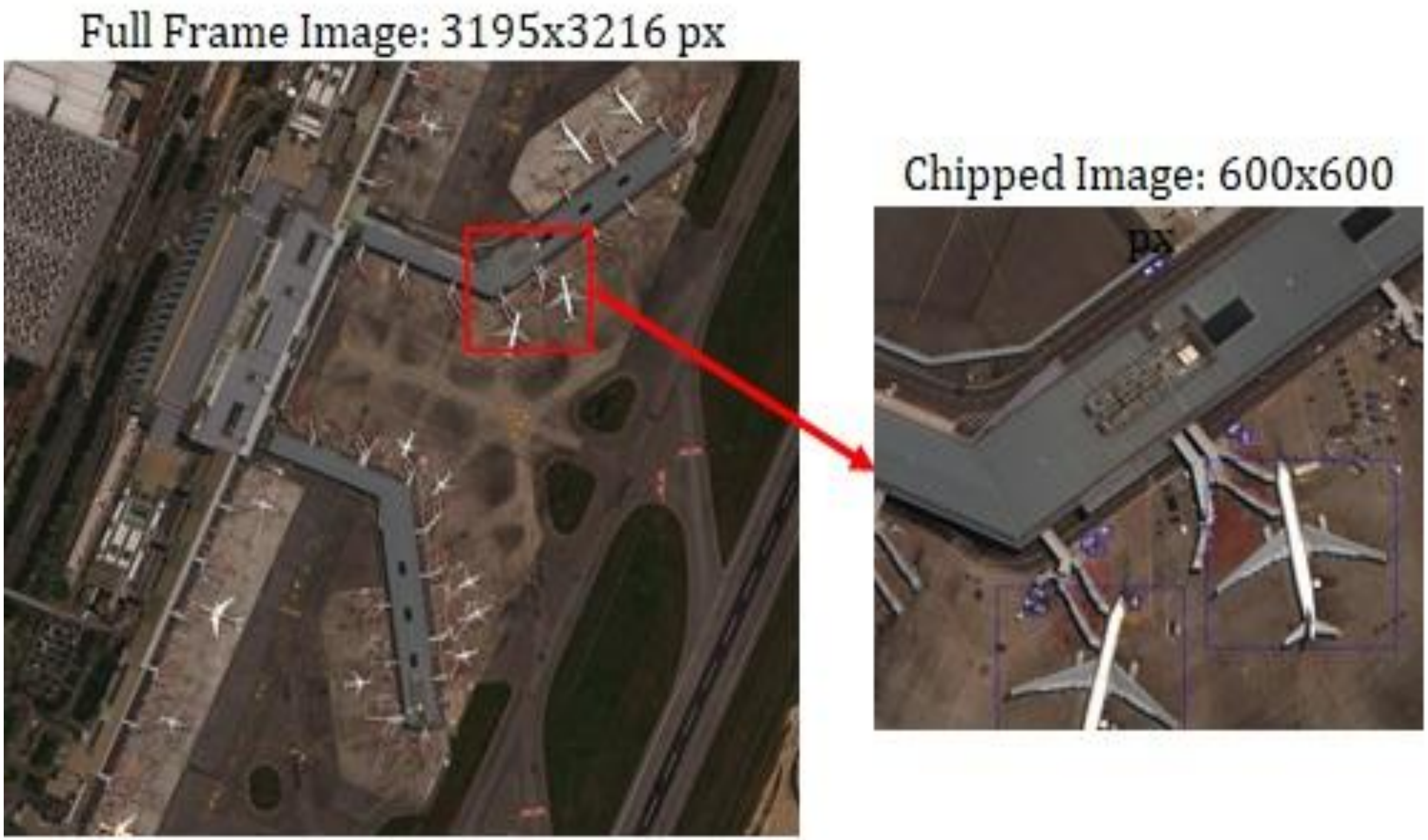
CNN Results – ℓ_1 min

Test data	Applied to SAR chips	Applied to ℓ_1 min results
Original Test Set	74.6%	85.6%
Corrupted Test Set	34.3%	60.7%
Noisy Test Set	17.3%	65.4%

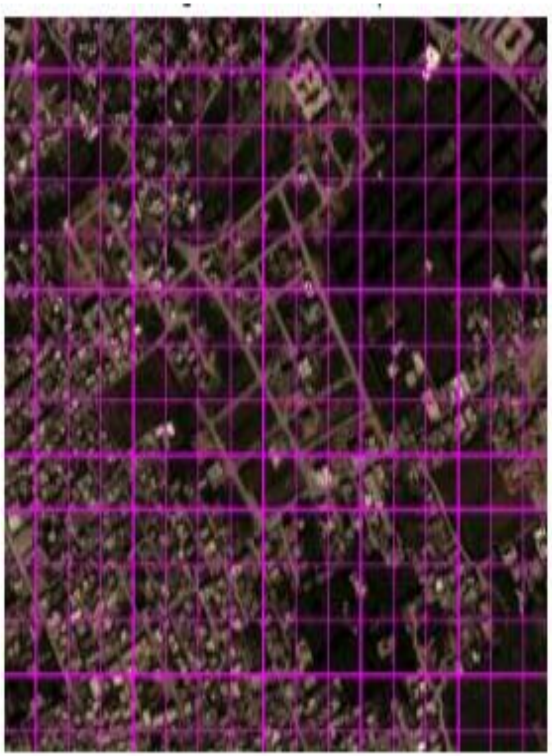
Multiple Targets: Challenges

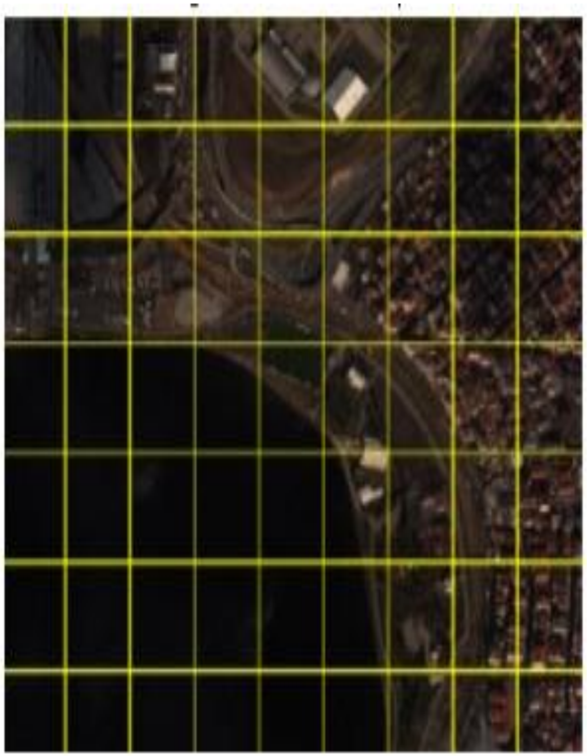
•**From** **http://xviewdataset.org**

Chipping and Labeling Targets from Large Scene



Detection Challenges: Homogeneous vs. Heterogeneous Targets Size





3.2 Classifier Specs

- **Custom DCNN**
  - 5 convolution/ReLU/pooling layers
  - 2 Fully Connected Layers
  - Trained for 100k iterations
- **Data**
  - 25 classes, including clutter class
  - # Training / Validation / Test Images: 191,788 / 63,929 / 63,930
  - Trained on translated images

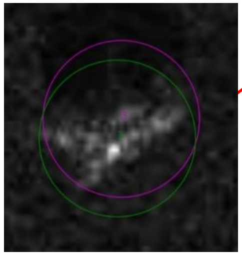
3.2 Classification Stage

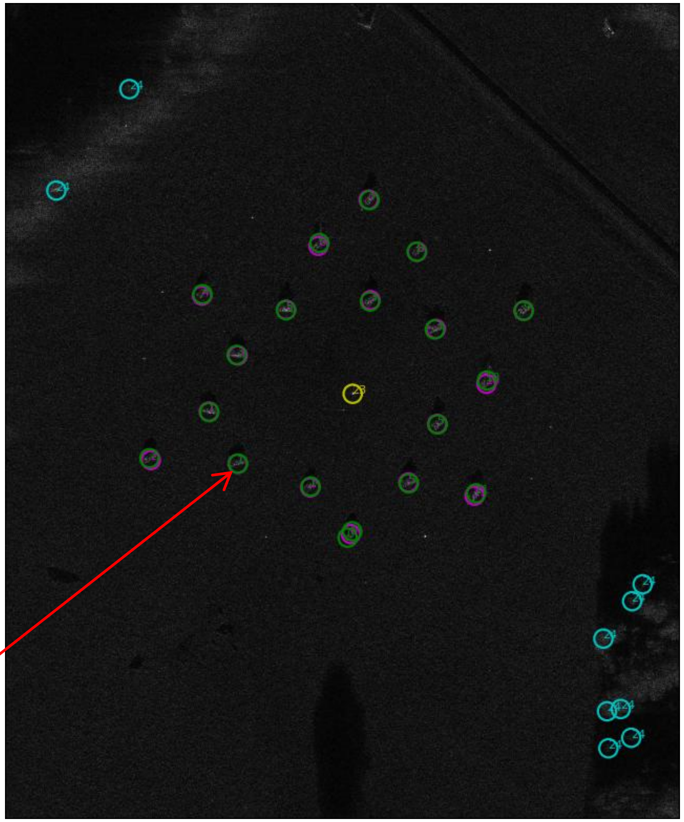
- **Result of detection stage is a directory containing the 128x128px image chips**
  - **Example: chip_330x827.jpg**
  - **(330,827) is the coordinate of the center pixel of the detected object in the image**
- **The classifier gets initialized then performs one forward pass (prediction) for each chip in the detection directory**
- **Results of classification are written to a text file of the format: "chip_name \t prediction \n"**

chip_330x827.jpg	22
chip_650x1102.jpg	4
…	

3.2 Example Result of Classification Task

- **Total Detections = 29**
- **100% of objects were detected**
- **# of misclassifications = 1 (yellow)**
- **Color of Circles:**
  - **Purple = GT objects**
  - **Green = correctly detected and classified objects**
  - **Blue = clutter that was detected but correctly classified as clutter**
  - **Yellow = clutter that was misclassified as an objects**
  - **Red = object that was detected but misclassified**





3.2 Conclusions on Multiple Target Classifications

- **We implemented Wavelet Decomposition to Reduce Clutter before CFAR Detection**
- **End-to-End CNN and CFAR-based multi-object classification**
- **Other Advanced Detection Techniques (such as R-CNN, Faster R-CNN, YOLO) techniques are used other than CFAR based Detection**
  - **Details on the RF ATR Textbook**