Machine Learning Techniques for Radar Automatic Target Recognition (ATR)

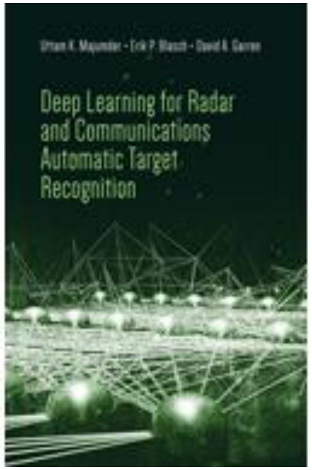
Chapter 10:
Recent/Emerging Topics in RF ATR

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:** **30 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:** **15 min**
9. **RF ATR Performance Evaluation:** **25 min**
10. **Emerging ML Algorithms for RF ATR: 35 min**



Emerging ML Algorithms for RF ATR

- **7 Habits of ATR**
- **Noise Induced / Adversarial Machine Learning**
- **Transfer Learning**
- **Active Learning**

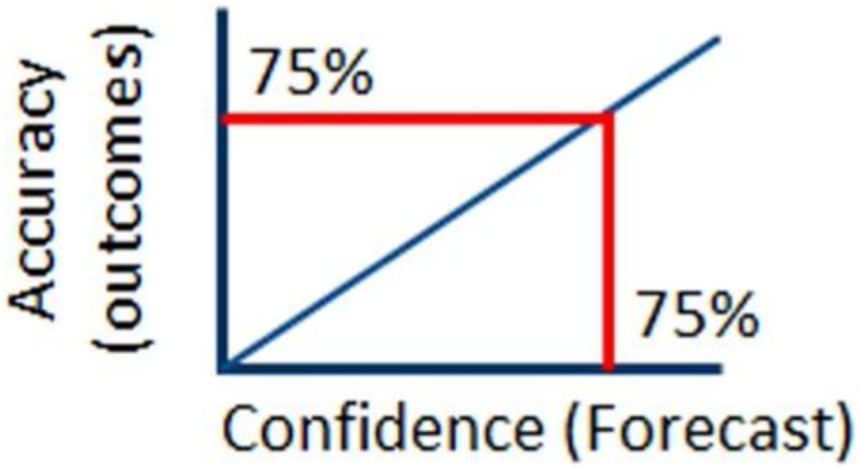
2.3 Seven Habits of Effective ATR

1. Confidence (How sure and reliable are you?)
2. Understandable (How does it work?)
3. Robust (Do you Gracefully Degrade under Variability?)
4. Out of Library Confusers (How effective are you?)
5. Performance Model (PM) (What is your Performance?)
6. Synthetic Data (How do you cover all Conditions?)
7. Sustainable End-to-End Training Process (How do you stay current?)

C. Paulson, L. Westerkamp, E. Zelnio., "Challenge problems consideration for the 7 habits of highly effective ATRs", (Conference Presentation)", Proc. SPIE 11393, Algorithms for Synthetic Aperture Radar Imagery XXVII, 113930V (27 April 2020); https://doi.org/10.1117/12.2561780

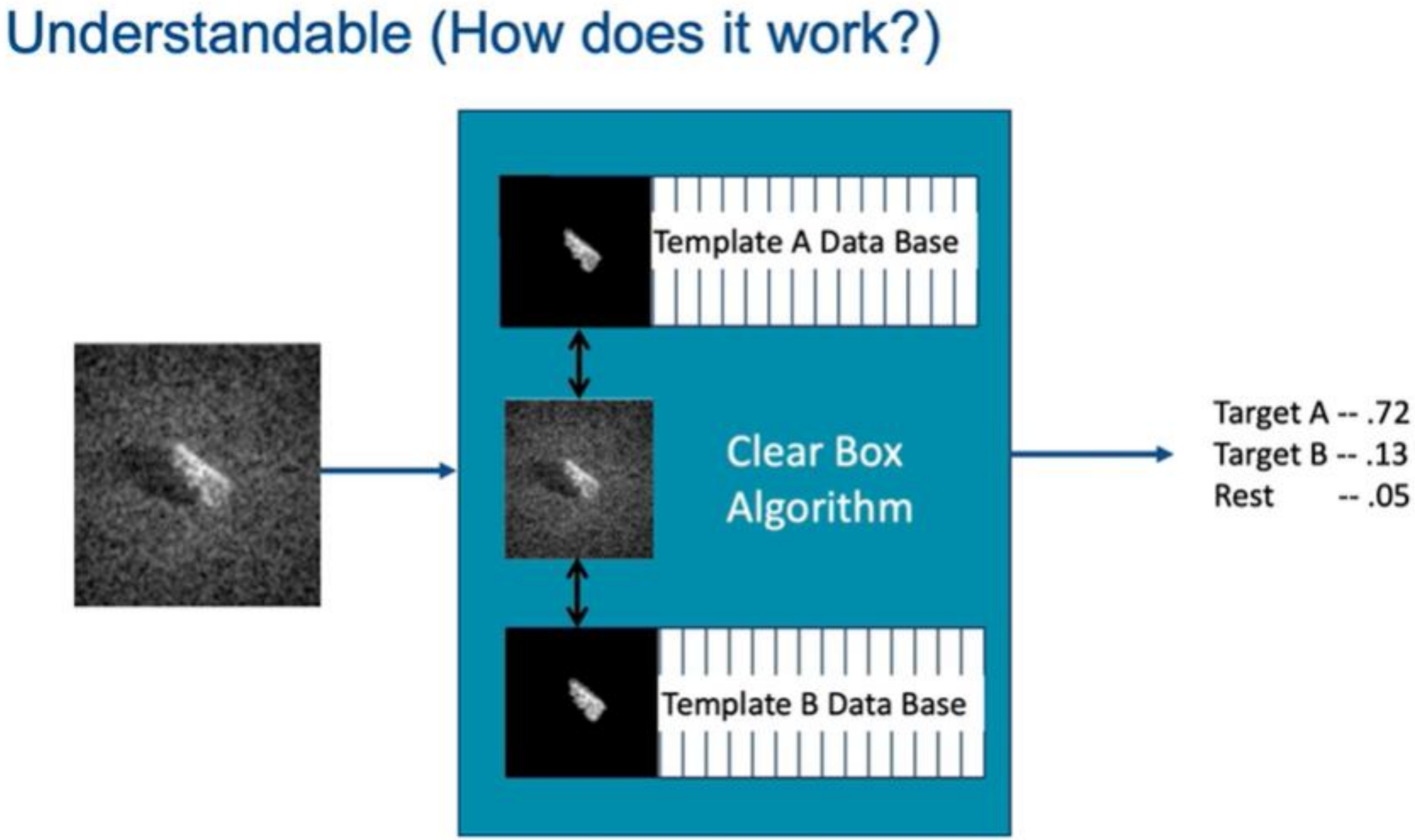
2.3.1 Confidence

Confidence (How sure and reliable are you?)



"And now the 7-day forecast..."

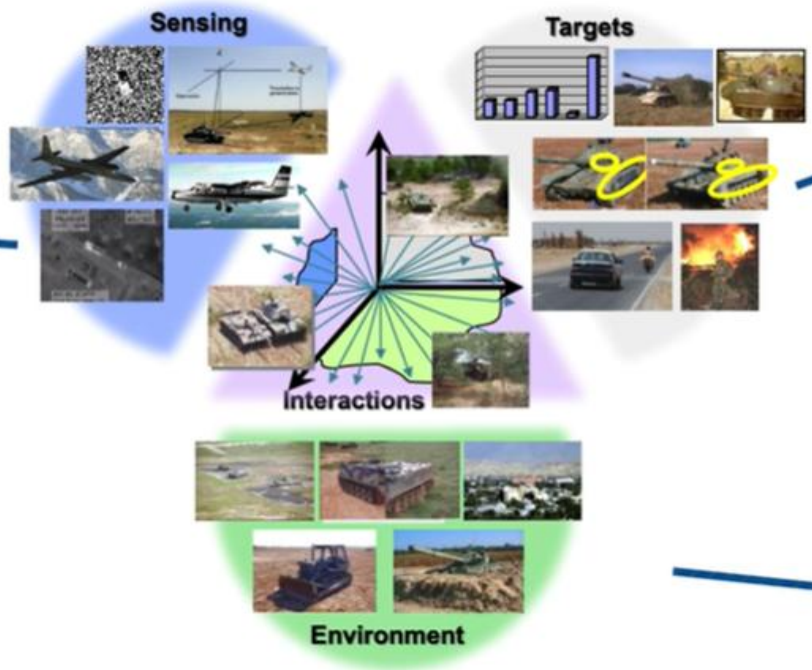
2.3.2 Understandable



2.3.3 Robust

Robust (Do you Gracefully Degrade under Variability?)

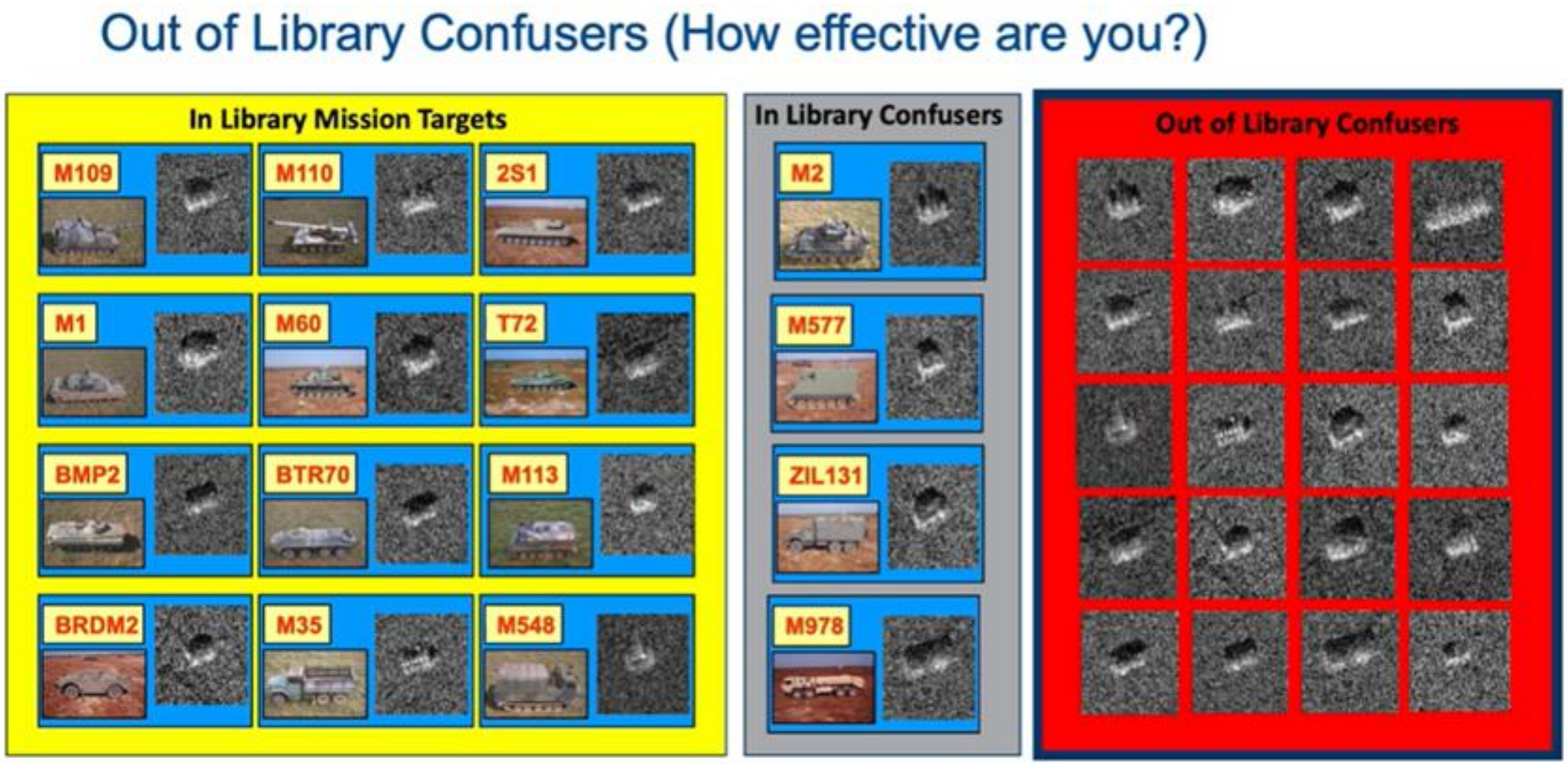
- Range
- Defocus
- Artifacts
- Calibration Errors



- Articulations
- Wear / Tear
- Extra / Missing Parts
- Configurations
- Adjacency

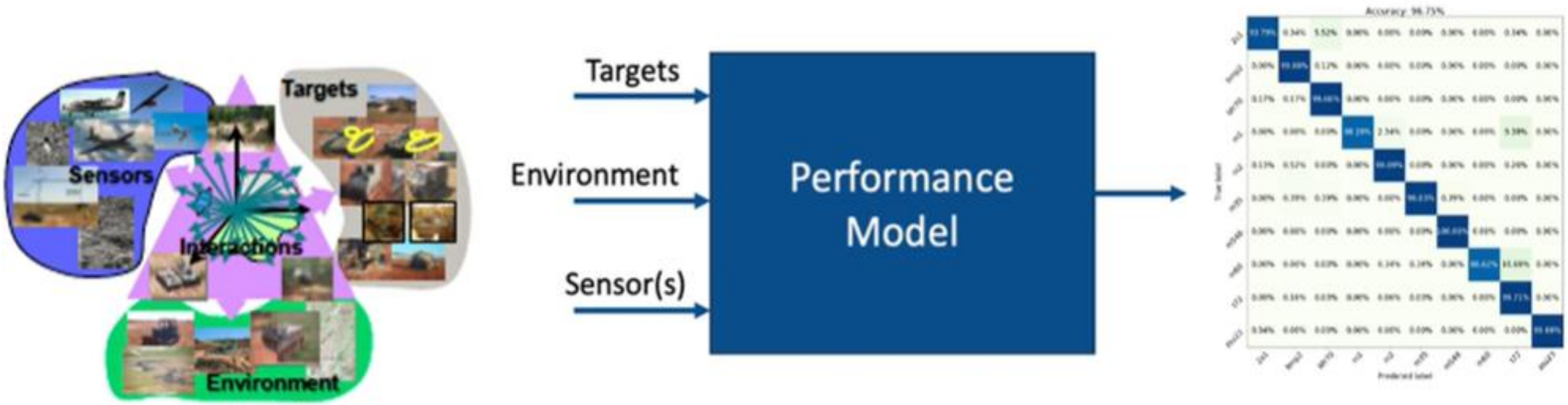
- Obscuration
- Atmosphere
- Sun Angle
- Terrain / Trees / Buildings

2.3.4. Out of Library / Distribution Confusers



2.3.5. ATR Performance Modeling

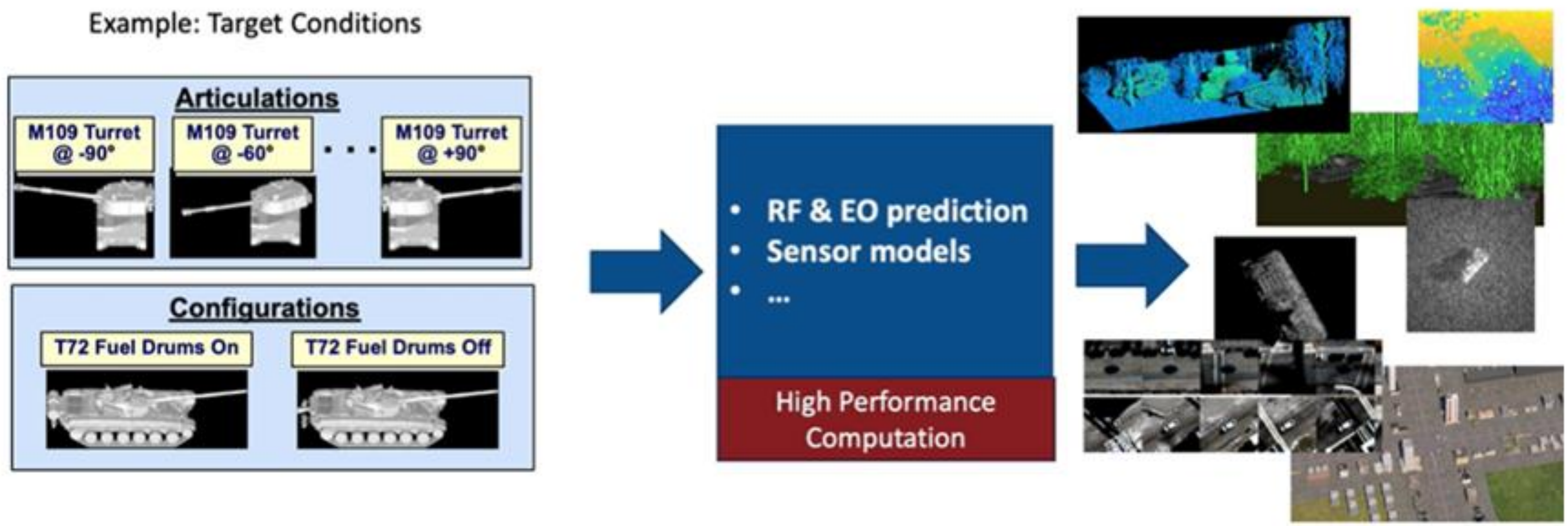
Performance Model (What is your Performance?)



2.3.6 Training with 100% Synthetic Data

Training with Synthetic Data (How do you cover all Conditions?)

- Limited Measured Data → Emerging Deep Learning Data Hungry
- Target, Environment, Sensor Conditions → Multi-Sensor Approaches
- Denied Targets



2.3.7 Real-time Training

Sustainable End-to-End Training Process (How do you stay current?)

- New Regions of Interest
- New Targets
- New Adversary Countermeasures
- Etc.

2.3.7 ATR Performance Evaluation Metric

$$\mathrm{PFA|Conf2} = \frac{\sum_{i=11}^{14}\sum_{j=1}^{10}\frac{(CM_{i,j})}{(CM_{i,15})}}{\sum_{i=11}^{14}\sum_{j=1}^{14}\frac{(CM_{i,j})}{(CM_{i,15})}} = \frac{\sum_{i=11}^{14}\sum_{j=1}^{10}\frac{(CM_{i,j})}{(CM_{i,15})}}{\sum_{i=11}^{14}1}$$	When given confusers (in library and out of library), what is the probability the system will declare a confuser as a mission target? This version normalizes the number of individual confusers.
$$\mathrm{PDec2} = \frac{\sum_{i=1}^{10}\sum_{j=1}^{10}\frac{(CM_{i,j})}{(CM_{i,15})}}{\sum_{i=1}^{10}\sum_{j=1}^{14}\frac{(CM_{i,j})}{(CM_{i,15})}} = \frac{\sum_{i=1}^{10}\sum_{j=1}^{10}\frac{(CM_{i,j})}{(CM_{i,15})}}{\sum_{i=1}^{10}1}$$	When given mission targets, what is the probability the system will report a mission target? This version normalizes the number of individual mission targets.
$$\mathrm{PID|Dec2} = \frac{\sum_{i=j=1}^{10}\frac{(CM_{i,j})}{(CM_{i,15})}}{\sum_{i=1}^{10}\sum_{j=1}^{10}\frac{(CM_{i,j})}{(CM_{i,15})}}$$	When given mission targets are declared, what is the probability the mission target was identified correctly? This version normalizes the number of individual mission targets.
$$\mathrm{PCC3} = \frac{\sum_{i=j=1}^{10}\frac{(CM_{i,j})}{(CM_{i,15})}}{\sum_{i=1}^{10}\sum_{j=1}^{10}\frac{(CM_{i,j})}{(CM_{i,15})} + \alpha\sum_{i=11}^{14}\sum_{j=1}^{10}\frac{(CM_{i,j})}{(CM_{i,15})}}$$	When mission targets are declared, what is the probability the mission target was identified correctly? This version normalizes the total number of confusers to the total number of mission targets, as well as the numbers of individual confusers and individual mission targets.

SAR ATR for Noisy Data

- When Signal and Images are corrupted with Noise and Clutter
  - **In Deep Learning Literature, it is called Adversarial Issues,**
    - **Adversarial Machine Learning (AML)**
  - **Research Goal:**
    - **How to tackle the Noise issue while Obtaining Accurate Classification**

88ABW-2019-5326
Nate Inkawhich, E. Davis, Uttam Majumder, et.al, "**Advanced Techniques for Robust SAR ATR:Mitigating Noise and Phase Errors**", IEEE Radar Conference 2020

4. Background & Idea

- **Background**
  - **Deep Learning has made SAR-ATR models very accurate in standard conditions**
  - **Interpretability of DL models is lacking**
  - **Robustness to many sources of noise is a concern**
- **Idea**
  - **If we train the network to be robust to worst-case noise, will robustness increase in general? Interpretability?**

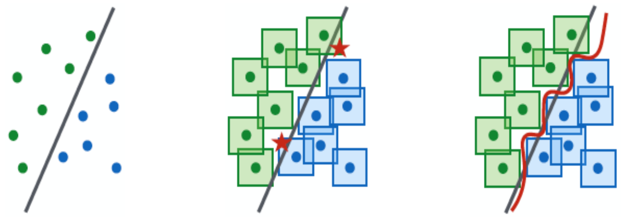
4. Adversarial Training

- Train the network parameters to minimize an "adversarial loss"

- Decision boundaries respect L_∞-norm balls around the training data

- Notable tradeoffs between clean data accuracy, model capacity, and dataset size have to be made

Intuition with L_∞ norm ball



Standard Training Objective

$$\min_{\theta} \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \left[L(x, y; \theta) \right]$$

Adversarial Training Objective

$$\min_{\theta} \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}} \left[\max_{\delta \in S} L(x+\delta, y; \theta) \right]$$

PGD Adversarial Attack

$$x^{t+1} = \Pi_{x+S}(x^t + \alpha sign(\nabla_{x^t} L(x^t, y; \theta)))$$

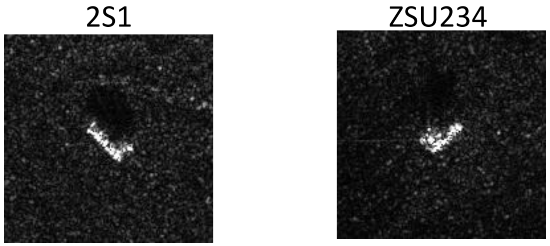
4. Dataset Used for this Research

MSTAR

- Collected by Sandia National Lab, funded by DARPA and AFRL

- X-band SAR sensor with 1-ft resolution spotlight mode and 360° aspect coverage

- 10 classes of targets

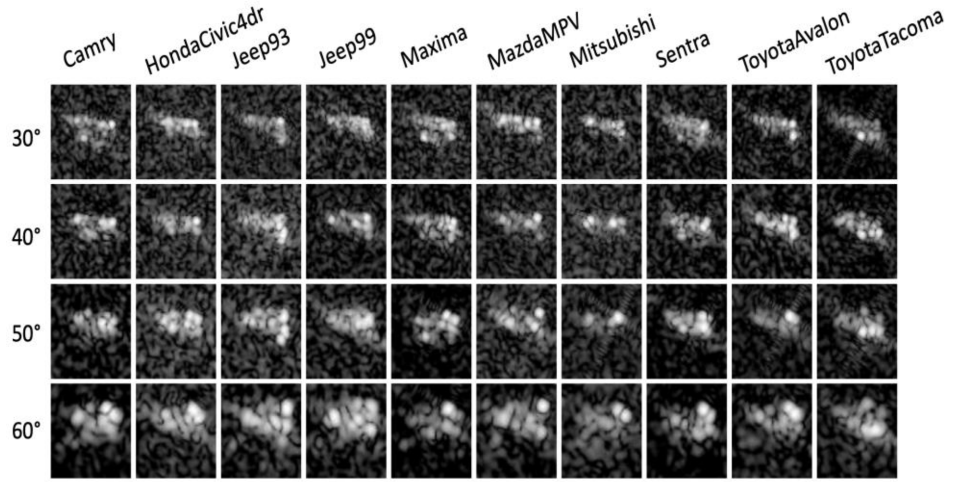


4. Civilian Vehicle Radar Data Domes (CVDome)

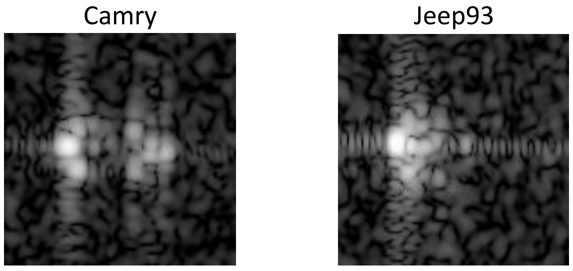
- **Simulated X-band phase history for 10 vehicle targets**

- **360° azimuth at elevations from 30° to 60°**

- **11x11 meter image chips generated at 0.3 meter resolution**



Polarization=HH, Azimuth=120



4. ATR Models

- **SAR ATR Community Models**
  - **A-ConvNet and ConvNetB**
- **Computer Vision Models**
  - **ResNet18, VGG11, ShuffleNetv2**
- **L_∞-norm**
  - **limit pixel-wise perturbation amount**
- **ε=0 → "standard" trained model**

TABLE I
DNN MODEL INFORMATION

model	lr	# params	MSTAR Acc	CVDome Acc
aconv	0.001	373,898	98.39	91.91
convb	0.001	9,512,970	98.54	90.96
rn18	0.1	111,753,370	97.57	95.67
vgg11	0.01	598,698	98.94	-
shuf	0.1	115,863	96.89	-

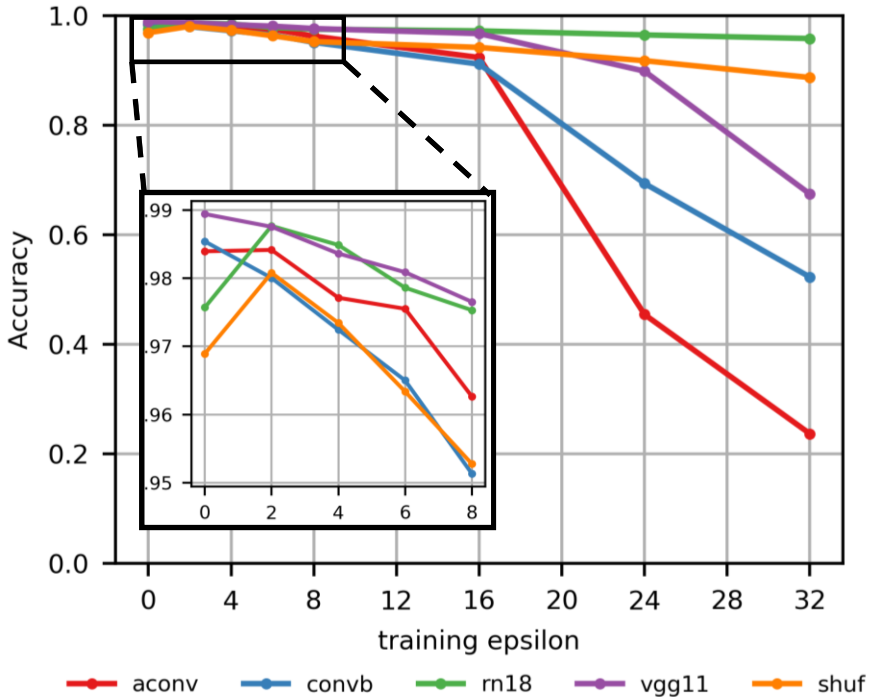
4. MSTAR Standard Operating Conditions (SOC)

- **Train: 17°, Test: 15° (Elevation)**

- **Accuracy degrades at large training *ε***
  - **Architecture dependent**

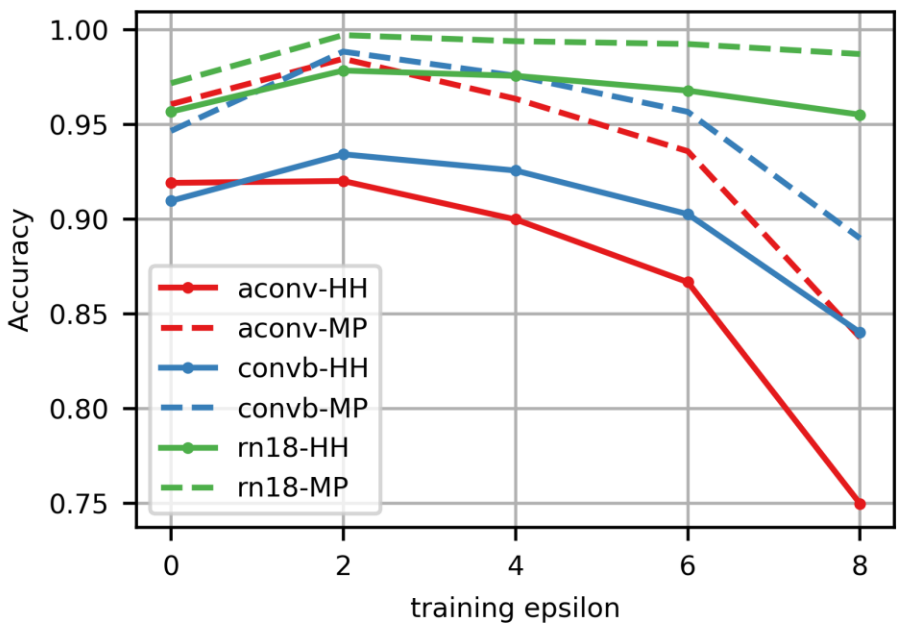
- **Some models have improved performance at small training *ε***

- **Takeaway: small training *ε*'s do not significantly harm MSTAR SOC performance**



4. CVDome Standard Operating Conditions

- Train: [30°, 40°, 50°, 60°]
- Test: [32°, 42°, 52°]
- Performance always improves with MP
- Some accuracy gain with small training ε
- Large accuracy loss in *aconv* and *convb* at $\varepsilon = 8$

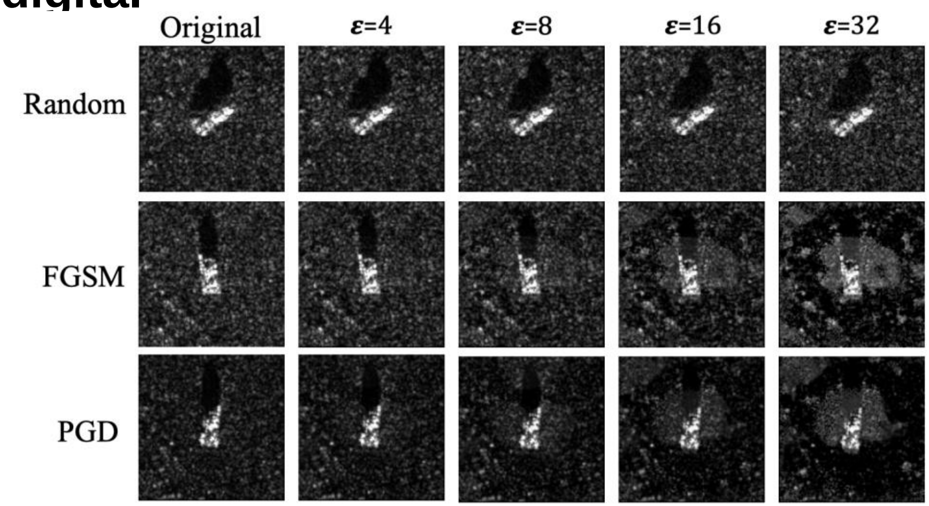


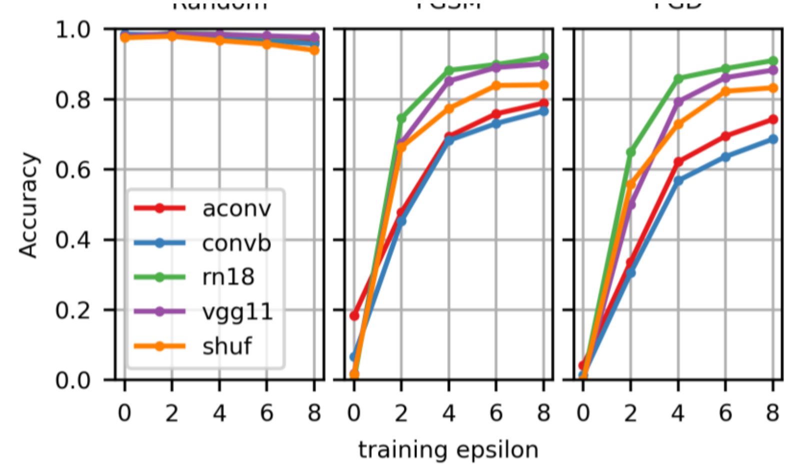
HH = single-polarization

MP = multi-polarization (HH,HV,VV)

4. Robustness: Adversarial Noise

- **Worst-case noise,** requires access to digital representation of data
- L_∞ noise becomes visible at ε>8
- Attacks completely degrade standard trained models
- AT recovers most accuracy loss
  - *rn18* highest performer
  - *aconv* and *convb* lowest perfor

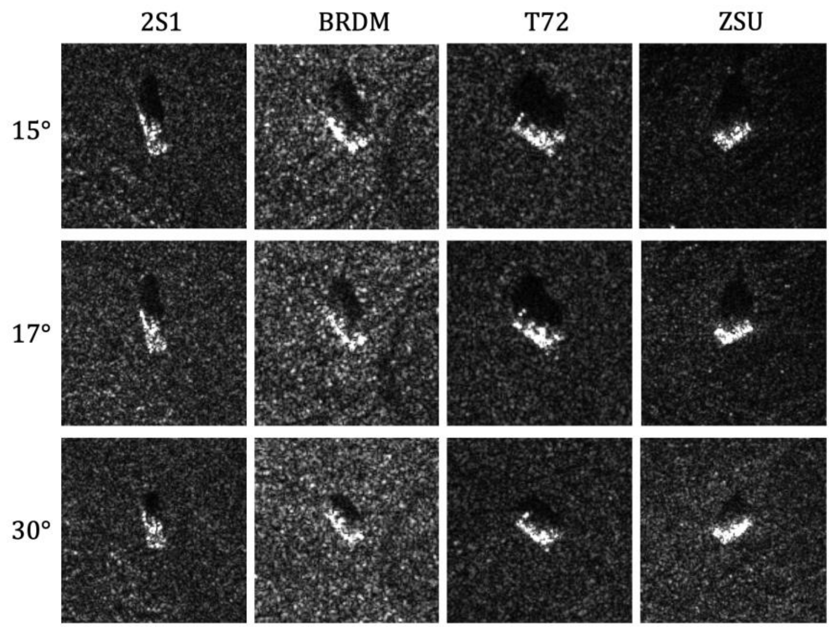




All test attacks generated at ε=8

4. Robustness: Extended Operating Conditions

- Train: 17°
- Test: 30°
- The 13° elevation shift causes changes in the shadow regions and target signatures
- Small training $\boldsymbol{\varepsilon}$ increases EOC performance
    - Rn18 $\boldsymbol{\varepsilon}$=2 increases accuracy by 11%



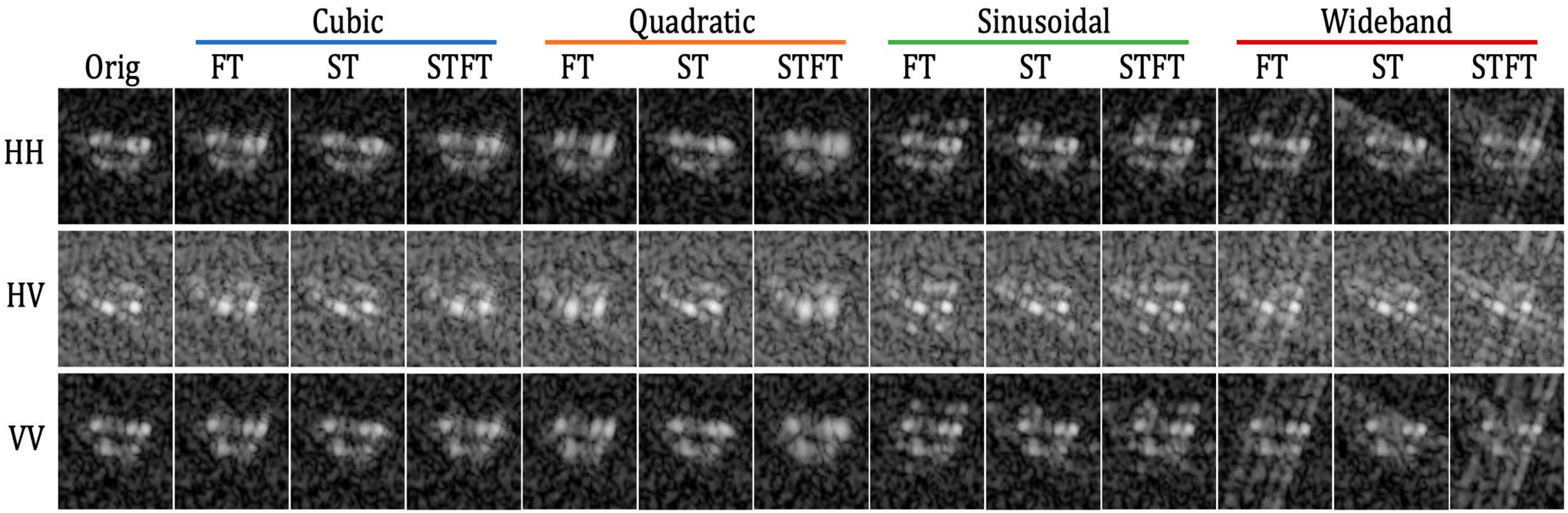
ACCURACY IN EXTENDED OPERATING CONDITIONS (MSTAR)

	training epsilon				
	0	2	4	6	8
aconv	82.2	85.1	85.2	84.7	84.6
convb	86.3	87.0	85.7	85.2	83.5
rn18	72.6	83.5	82.7	82.7	82.6
shuf	75.8	81.6	81.7	81.3	80.2
vgg11	69.3	84.4	82.9	82.9	81.8

4. Robustness: Phase Errors

- Add cubic, quadratic, sinusoidal, and wideband noise to the CVDome phase history data
- Noise added in the range (fast time = FT) and/or azimuth (slow time = ST) dimensions

$$s_e[n,p] = s[n,p]e^{(i(\phi_{(e,FT)}[n]+\phi_{(e,ST)}[p])}$$



4. Robustness: Phase Errors

- **Sinusoidal and wideband noise are the most challenging**
- **Rn18 is highest performing architecture**
- **MP increases robustness over HH**
- **AT increases robustness, in many cases by over 20%**

ROBUSTNESS TO PHASE ERRORS (CVDOME)

model	eps	noError	cubicFT	cubicST	cubicSTFT	quadraticFT	quadraticST	quadraticSTFT	sinusoidalFT	sinusoidalST	sinusoidalSTFT	widebandFT	widebandST	widebandSTFT
aconv (HH)	0	91.91	87.84	89.7	83.79	81.97	87.98	73.93	57.35	87.25	44.2	53.79	78.64	33.47
	2	92.02	89.29	90.56	85.62	83.25	88.78	73.66	60.46	88.66	50.74	56.83	81.51	40.41
	4	89.98	88.23	88.05	84.19	80.21	87.03	72.47	59.05	87.23	52.24	58.5	80.38	44.76
	6	86.67	85.53	85.46	81.67	77.26	84.61	69.69	57.52	84.62	52.33	56.52	78.46	46.5
	8	74.97	73.01	73.05	67.71	63.92	71.38	56.02	51.35	73.34	49.32	52.16	67.96	46.67
convb (HH)	0	90.96	87.34	88.79	83.25	82.41	87.05	71.53	59.87	86.44	46.8	50.92	78.78	29.32
	2	93.42	90.87	91.6	86.53	84.84	89.35	73.76	62.39	89.98	52.51	56.64	82.66	36.69
	4	92.56	90.41	90.21	85.6	84.12	89.14	73.91	62.38	89.55	54.2	57.32	82.07	40.17
	6	90.26	88.1	88.55	84.01	82.66	87.02	72.89	60.37	87.46	52.2	56.62	79.03	42.15
	8	84.01	81.44	82.03	76.82	75.06	79.74	64.52	55.79	80.02	50.51	54.24	71.01	45.06
rn18 (HH)	0	95.67	89.89	92.65	82.11	85.56	90.75	71.83	62.7	89.62	48.56	51.26	77.24	27.61
	2	97.84	95.79	96.34	92.28	90.97	93.66	79.34	63.69	93.62	55.08	64.71	86.83	44.97
	4	97.56	95.17	96.16	92.41	90.01	92.71	78.69	62.35	94.33	54.84	66.98	87.7	53.91
	6	96.78	93.57	95.07	90.73	87.98	91.08	75.56	61.15	93.91	53.87	67.67	87.15	55.39
	8	95.51	92.41	94.11	89.42	86.47	89.93	74.39	59.25	92.98	54.55	66.64	86.79	58.14
aconv (MP)	0	96.06	93.61	95.67	91.06	87.28	94.65	78.98	75.52	93.62	61.52	64.73	87.96	42.03
	2	98.46	96.83	97.92	94.97	91.26	97.16	85.08	77.34	97.2	67.93	73.55	93.74	55.85
	4	96.35	94.41	95.51	92.37	87.6	94.75	81.64	74.28	95.96	69.84	74.15	92.92	61.39
	6	93.58	91.23	92.74	88.91	83.39	91.67	76.74	71.02	93.08	67.5	73.02	89.89	63.35
	8	83.74	82.12	83.03	79.2	72.98	82.21	68.97	62.39	82.69	61.	63.38	80.8	58.15
convb (MP)	0	94.65	93.82	94.43	91.07	87.73	93.78	78.79	80.08	92.53	65.74	65.35	87.89	40.12
	2	98.84	97.88	98.56	96.3	93.79	97.71	86.15	82.98	98.21	74.07	77.26	95.6	57.03
	4	97.56	96.32	97.43	94.87	92.83	96.42	85.78	80.39	97.43	74.74	77.75	95.06	61.74
	6	95.66	94.16	95.05	92.87	90.1	94.05	83.58	77.08	95.67	72.55	73.5	92.73	62.12
	8	88.98	87.01	87.87	85.15	79.67	86.84	75.26	66.41	87.83	63.89	65.66	84.42	60.05
rn18 (MP)	0	97.17	96.37	96.82	94.2	93.61	96.42	87.62	83.	95.24	69.6	67.14	87.47	37.03
	2	99.71	99.16	99.35	98.39	97.23	98.75	92.46	82.01	98.38	72.26	83.12	95.94	65.47
	4	99.39	98.75	99.01	98.07	97.1	98.2	92.21	81.42	98.44	74.01	87.55	96.69	74.52
	6	99.24	98.56	98.67	97.83	96.26	97.69	90.5	78.42	98.06	72.05	87.29	96.47	77.
	8	98.71	97.88	98.25	97.02	95.16	96.87	89.01	76.16	97.55	71.23	86.57	95.82	78.73

4. Robustness: Phase Errors

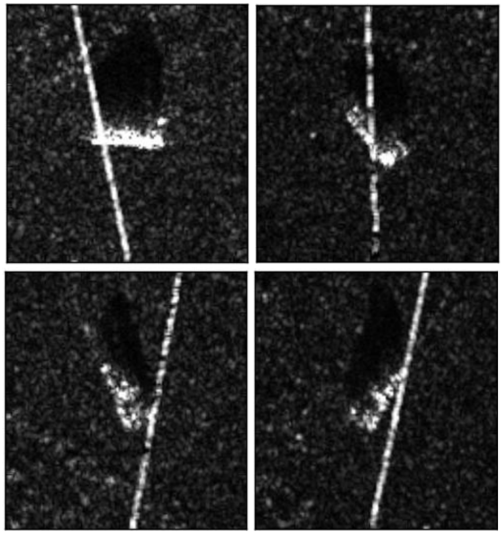
- **Sinusoidal and wideband noise are the most challenging**
- **Rn18 is highest performing architecture**
- **MP increases robustness over HH**
- **AT increases robustness, in many cases by over 20%**

ROBUSTNESS TO PHASE ERRORS (CVDOME)

model	eps	noError	cubicFT	cubicST	cubicSTFT	quadraticFT	quadraticST	quadraticSTFT	sinusoidalFT	sinusoidalST	sinusoidalSTFT	widebandFT	widebandST	widebandSTFT
aconv (HH)	0	91.91	87.84	89.7	83.79	81.97	87.98	73.93	57.35	87.25	44.2	53.79	78.64	33.47
	2	92.02	89.29	90.56	85.62	83.25	88.78	73.66	60.46	88.66	50.74	56.83	81.51	40.41
	4	89.98	88.23	88.05	84.19	80.21	87.03	72.47	59.05	87.23	52.24	58.5	80.38	44.76
	6	86.67	85.53	85.46	81.67	77.26	84.61	69.69	57.52	84.62	52.33	56.52	78.46	46.5
	8	74.97	73.01	73.05	67.71	63.92	71.38	56.02	51.35	73.34	49.32	52.16	67.96	46.67
convb (HH)	0	90.96	87.34	88.79	83.25	82.41	87.05	71.53	59.87	86.44	46.8	50.92	78.78	29.32
	2	93.42	90.87	91.6	86.53	84.84	89.35	73.76	62.39	89.98	52.51	56.64	82.66	36.69
	4	92.56	90.41	90.21	85.6	84.12	89.14	73.91	62.38	89.55	54.2	57.32	82.07	40.17
	6	90.26	88.1	88.55	84.01	82.66	87.02	72.89	60.37	87.46	52.2	56.62	79.03	42.15
	8	84.01	81.44	82.03	76.82	75.06	79.74	64.52	55.79	80.02	50.51	54.24	71.01	45.06
rn18 (HH)	0	95.67	89.89	92.65	82.11	85.56	90.75	71.83	62.7	89.62	48.56	51.26	77.24	27.61
	2	97.84	95.79	96.34	92.28	90.97	93.66	79.34	63.69	93.62	55.08	64.71	86.83	44.97
	4	97.56	95.17	96.16	92.41	90.01	92.71	78.69	62.35	94.33	54.84	66.98	87.7	53.91
	6	96.78	93.57	95.07	90.73	87.98	91.08	75.56	61.15	93.91	53.87	67.67	87.15	55.39
	8	95.51	92.41	94.11	89.42	86.47	89.93	74.39	59.25	92.98	54.55	66.64	86.79	58.14
aconv (MP)	0	96.06	93.61	95.67	91.06	87.28	94.65	78.98	75.52	93.62	61.52	64.73	87.96	42.03
	2	98.46	96.83	97.92	94.97	91.26	97.16	85.08	77.34	97.2	67.93	73.55	93.74	55.85
	4	96.35	94.41	95.51	92.37	87.6	94.75	81.64	74.28	95.96	69.84	74.15	92.92	61.39
	6	93.58	91.23	92.74	88.91	83.39	91.67	76.74	71.02	93.08	67.5	73.02	89.89	63.35
	8	83.74	82.12	83.03	79.2	72.98	82.21	68.97	62.39	82.69	61.	63.38	80.8	58.15
convb (MP)	0	94.65	93.82	94.43	91.07	87.73	93.78	78.79	80.08	92.53	65.74	65.35	87.89	40.12
	2	98.84	97.88	98.56	96.3	93.79	97.71	86.15	82.98	98.21	74.07	77.26	95.6	57.03
	4	97.56	96.32	97.43	94.87	92.83	96.42	85.78	80.39	97.43	74.74	77.75	95.06	61.74
	6	95.66	94.16	95.05	92.87	90.1	94.05	83.58	77.08	95.67	72.55	73.5	92.73	62.12
	8	88.98	87.01	87.87	85.15	79.67	86.84	75.26	66.41	87.83	63.89	65.66	84.42	60.05
rn18 (MP)	0	97.17	96.37	96.82	94.2	93.61	96.42	87.62	83.	95.24	69.6	67.14	87.47	37.03
	2	99.71	99.16	99.35	98.39	97.23	98.75	92.46	82.01	98.38	72.26	83.12	95.94	65.47
	4	99.39	98.75	99.01	98.07	97.1	98.2	92.21	81.42	98.44	74.01	87.55	96.69	74.52
	6	99.24	98.56	98.67	97.83	96.26	97.69	90.5	78.42	98.06	72.05	87.29	96.47	77.
	8	98.71	97.88	98.25	97.02	95.16	96.87	89.01	76.16	97.55	71.23	86.57	95.82	78.73

4. Robustness: Radio Interference (RI)

- RI noise appears as strong stripes in the range dimension
- Standard model accuracy around 20% lower than SOC because stripe directly alters representation
- Noise impacts each architecture differently:
  - For *aconv* and *convb*, AT harms robustness
  - For *rn18*, AT boosts robustness by 2.6%



ROBUSTNESS TO INTERFERENCE NOISE (MSTAR)

	training epsilon				
	0	2	4	6	8
aconv	71.0	72.9	72.1	73.1	71.7
convb	77.5	73.9	73.0	72.4	72.7
rn18	80.4	80.8	**83.0**	**83.0**	82.2
shuf	71.5	55.8	60.1	58.7	57.1
vgg11	79.1	80.9	77.5	79.7	75.9

Summary on Noise Induced SAR/RF ATR

- Robustness and interpretability improve with AT
- AT models do not rely on shadow regions
- Multi-polarization information helpful in accuracy and robustness
- The impact of AT greatly depends on architecture
  - **Small and fast models not good for AT**
  - **Rn18 works the best here but is the largest**

Transfer Learning Research Goal

- **Transfer Learning (TL) is an important research area of Machine Learning when *<u>measured data are limited</u>* OR Training and Testing data *<u>do not match</u>* (sensor variations, sensitivity of the data collection system etc.)**
- **Often measured radar data are expensive due to cost associated with the collection**
- **Synthetic radar data generation is an inexpensive way of gathering radar imagery (for ML research)**
- **However, synthetic radar data are pristine and do not resemble measured data well**
- **<u>Hence Our Research Problem</u>: *Investigating Technical Approaches to Develop a DNN model with 100% Synthetic Data and Test with Measured Data while Achieving Better Than SoTA Accuracy***
  - **We achieved 97% Max. accuracy (92% Avg.) compared to 25% presented in SAMPLE dataset challenge problem.**
  - **Other researcher achieved 95% accuracy while using (99% synthetic and 1% measured data). They did not use 100% synthetic data**

AFRL SAMPLE Dataset

- **AFRL/RY published <u>Synthetic and Measured Paired Labeled Experiment (SAMPLE)</u> dataset for Transfer Learning research**
- **The SAMPLE dataset was constructed by simulating radar capture of CAD models of the MSTAR dataset.**
- **The generation of the CAD models included human input to ensure that the models matched the vehicles.**
- **The simulated radar capture was done matching the angle and other metadata of the images in the MSTAR dataset.**

Benjamin Lewis, Theresa Scarnati, Elizabeth Sudkamp, John Nehrbass, Stephen Rosencrantz, and Edmund Zelnio "A SAR dataset for ATR development: the Synthetic and Measured Paired Labeled Experiment (SAMPLE)", Proc. SPIE 10987, Algorithms for Synthetic Aperture Radar Imagery XXVI, 109870H (14 May 2019)

Challenge Problems Using SAMPLE Dataset

 # 1: Classifying Measured Data from 100% Synthetic Training Data

-- This is the research we are presenting here

 # 2: For 10 Targets type, say 8 have been trained with measured and 2 have been trained with synthetic data. After testing with measured dataset, analyze performance of all 10 targets type, 2 synthetic targets and 8 measured targets.

 # 3: The Open Set Problem: Out of 10 target types, remove training data for 2 target types. Then take a fraction of synthetic and measured data to train 8 classes. Now, analyze performance of 10 classes on measured data

Transfer Learning

- **The filed of Transfer Learning has been expanded and hence it can be of various types (Interested reader may look into "A Survey on Transfer Learning")**
- **In Radar sensor, TL is most often used to work on Synthetic and Measured radar data (In this research). It may also include Radar frequency changes (X-band and C-band/Ku-Band) on train/test**
- **TL may include training on Electro-optical (video imagery) data but testing on radar data**
- **"Domain Adaptation" is often used for TL**
- **Few-shot Learning / One-shot Learning**

S. J. Pan and Q. Yang, "A Survey on Transfer Learning," in *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345-1359, Oct. 2010, doi: 10.1109/TKDE.2009.191.

Transfer Learning on SAMPLE data

SAMPLE Challenge problem paper shows that when 100% synthetic data are used with a very simple DNN model, it achieves less than 50% accuracy when tested on the measured data. We use this as a starting point and use various methods to improve performance from that model.

DNN Models and Hyperparameter

We tested the following DNN hyperparameters to examine their impact on classification accuracy:

- **Rotation**
- **Uniformly distributed random noise**
- **Gaussian random noise**
- **Simulated radar clutter**
- **Horizontal flip**
- **Learning rate decay**
- **Dropout**
- **Weight decay**

We tested the following model types:

- Model defined by the SAMPLE paper
- Resnet18
- Resnet18 wide
- Resnet50
- Densenet121

Mixup

Mixup is a combination of two randomly selected images within the training set. Using the Beta distribution to generate a value $\lambda \in [0,1]$. Each pixel of the mixup, is defined by $p_{\{i,m\}}=\lambda*p_{\{i,1\}} + (1-\lambda)*p_{\{i,2\}}$ where $p_{\{i,1\}}$ represents the *i*th pixel in the first image and $p_{\{i,2\}}$ represents the *i*th pixel in the second image.

The two images may be from different classes, in which case the new label to be used will be the one hot vector with the value for image 1's class as λ and the value for image 2's class as $1-\lambda$

Mixup (cont.)

We tried two methods for adding mixup:

- **Mixup within each batch - given permutation P, index i, within each batch is replaced by the mixup of i and P[i]**
- **Mixup augmenting the dataset - For each datum to be added, two random data are selected and the mixup of those two data are added to the dataset in addition to the rest of the dataset.**

Cosine Loss

- **Cosine loss is an alternative to SGD that is based solely on vector direction instead of magnitude.**
- **It restricts the domain to the unit hypersphere by normalizing the feature space. This bounds the output to [0,2] and ignores any lack of regularization which may be present.**
- **These both help when training on small datasets like SAMPLE, especially because the synthetic data does have a different average brightness than the measured data.**

Label Smoothing

- **Changes each integer class label into a one hot vector with other classes represented by the uniform random distribution**

- **Keeps the model from becoming too sure of its classifications**

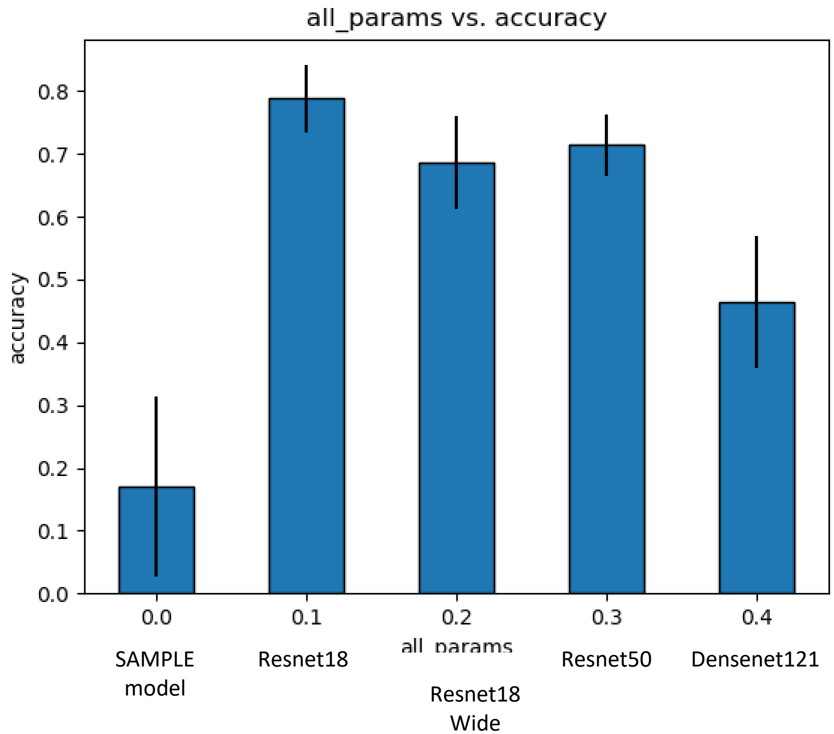
Optimal Parameters Determined Individually

- Iterations = 40
- Batch size = 16
- learning rate = 10^-3
- Gaussian noise = .4
- Uniform noise = 0
- Mixup alpha = .8
- Weight decay = 10^-4

- Rotation = 20
- Flip = .1
- Dropout = .2*
- sim clutter = .8
- learning rate decay = .4 after 30 epochs
- label smoothing = .1
- cosine loss

Results

Different models with naively selected optimal parameters



Optimal Parameters on Resnet18

- Iterations = 40 (Some testing done on 100)
- Batch size = 16 (Some testing done on 64)
- learning rate = 10^-3
- Gaussian noise = .4
- Uniform noise = 0
- Mixup alpha = 0
- Weight decay = 0
- Rotation = 0
- Flip = 0
- Dropout = .4
- Sim clutter = 0
- Learning rate decay = 0
- Label smoothing = .1
- Cross entropy loss

Gaussian + Dropout + OTHER (dsize=64, b_size=128, norm=[-1,1], #trains=100, epochs=60, lr=0.001)					
Method	Min-Acc	Max-Acc	Avg-Acc	std-Acc	Perf-Know
Model = sample_model					
gaussian_std=0.3, drop=0.3	0.7087	0.9072	0.8207	0.0414	0.9202
gaussian_std=0.3, drop=0.3, label_smooth=0.08	0.7643	0.935	0.8669	0.0341	0.9462
gaussian_std=0.3, drop=0.3, mixup=0.1	0.6586	0.9183	0.8416	0.0431	0.9257
gaussian_std=0.3, drop=0.3, cosine_loss	0.6103	0.922	0.8334	0.051	0.9462
gaussian_std=0.2, drop=0.2, AT (eps=2, alpha=0.5, its=7)	0.7866	0.9276	0.8645	0.0289	0.9387
gaussian_std=0.2, drop=0.2, AT (eps=4, alpha=1, its=7)	0.7495	0.9276	0.8639	0.0375	0.9332
gaussian_std=0.2, drop=0.2, AT (eps=8, alpha=2, its=7)	0.8107	0.9369	0.8851	0.0268	0.948
Model = resnet18					
gaussian_std=0.4, drop=0.4	0.8126	0.9536	0.8983	0.03	0.9628
gaussian_std=0.4, drop=0.4, label_smooth=0.08	0.846	0.9536	0.9119	0.0233	0.974
gaussian_std=0.4, drop=0.4, label_smooth=0.1	0.8497	0.9554	0.9187	0.0217	0.9647
gaussian_std=0.4, drop=0.4, mixup=0.1	0.8441	0.9684	0.9079	0.0264	0.9684
gaussian_std=0.4, drop=0.4, cosine_loss	0.8293	0.9573	0.9041	0.0266	0.961
gaussian_std=0.3, drop=0.3, AT (eps=2, alpha=0.5, its=7)	0.8311	0.948	0.9022	0.0263	0.9666
gaussian_std=0.3, drop=0.3, AT (eps=4, alpha=1, its=7)	0.8627	0.9443	0.9012	0.0204	0.9703
gaussian_std=0.3, drop=0.3, AT (eps=8, alpha=2, its=7)	0.8478	0.9387	0.893	0.0199	0.9499
AT16					
Model = wide-resnet18					
gaussian_std=0.4, drop=0.4	0.8051	0.9461	0.8879	0.0285	0.9628
gaussian_std=0.4, drop=0.4, label_smooth=0.08	0.8497	0.9591	0.9105	0.0241	0.9703
gaussian_std=0.4, drop=0.4, mixup=0.1	0.794	0.9721	0.9014	0.0321	0.974
gaussian_std=0.4, drop=0.4, cosine_loss	0.7977	0.9536	0.9001	0.0292	0.9684
gaussian_std=0.3, drop=0.3, AT (eps=2, alpha=0.5, its=7)	0.8144	0.9536	0.9043	0.0281	0.9628
gaussian_std=0.3, drop=0.3, AT (eps=4, alpha=1, its=7)	0.8367	0.9517	0.8966	0.0283	0.9628
gaussian_std=0.3, drop=0.3, AT (eps=8, alpha=2, its=7)	0.8237	0.9369	0.8852	0.0246	0.9554

Some of the parameter compositions we tried searching for the maximum accuracy

Summary on Transfer Learning

- **DNN Models Play a big role on TL Performance**
  - ✓ **Resnet18 Performs better than any other Model**
- **Adding Gaussian Noise (0.4) to the Synthetic Data for Training boost Performance**
- **Dropout (0.4) and Label Smoothing (0.1) also improved classification performance**
- **Overall, we achieved 92% (Avg.) accuracy using 100% synthetic Data for training and testing on measured data.**
  - ✓ **This is huge and first of it’s kind result on the SAMPLE dataset**
- **Other approaches (Ensembles) can be incorporated to improve the accuracy further**

5. Future Research Challenges: RF SAR ATR

- **Transfer Learning / Domain Adaptation**
  - 100% Synthetic Data for Training, measured Data for Testing
- **Out of Distribution Detection and Classification**
  - Reducing False Alarm in the Presence of Confusers
- **Real-time Training**
- **Sufficient Statistic Analysis for Object Classification**
  - Least amount of data needed for Training yet achieving high accuracy
- **On-Chip, On-line, Power Efficient Learning Hardware**
  - ***Memristor, Neuromorphic Computing, Intel Loihi (Spiking Neural Networks)***
- **Multiple-int (RF, EO) Fusion for Target Recognition**
- **Quantum ML algorithms for RF target classification**
- **SAR ATR Complexity Analysis**

Thank you.