SAMPLE with a Side of MSTAR: Extending SAMPLE with Outliers and Target Variants from MSTAR

Benjamin Lewis[a], Mark Ashby[a], and Edmund Zelnio[a]

[a]Air Force Research Laboratory, Sensors Directorate, WPAFB, OH

ABSTRACT

Deep learning is a technology applied to a host of problems in the decade since its introduction. Of particular interest for both defense and civil applications is the technology of automatic target recognition, which is a subset of visual detection and classification. However, these classification algorithms must be robust to out-of-library confusers and able to generalize across a variety of target types. In this paper, we augment the existing Synthetic and Measured Paired Labeled Experiment dataset of synthetic aperture radar images with the remainder of the public MSTAR dataset and define a set of experiments to encourage the development of traits beyond simple classification accuracy for target recognition algorithms.

Keywords: SAMPLE dataset, SAR, Dataset, MSTAR, Challenge Problem, Out-of-domain Learning

1. INTRODUCTION

The world of machine learning has had an exciting decade since the introduction of AlexNet[1] and the popularization of convolutional neural networks (CNNs) in 2012. Deep learning has enabled a variety of applications in image recognition, language translation, computer-generated text, robotics, and many other fields. Impressive new results are presented almost daily in this field of endeavor. Naturally, individual researchers must specialize in only a small portion of the wide field of machine learning models. In our case, we are most interested in the applications of computer vision models to synthetic aperture radar (SAR) imagery for target identification.

With the capability of these complex, black-box vision models comes the natural question of how robust they are to the real world, both in the variety of configurations objects can appear in as well as how the model handles the entire world of objects it was not trained on. These problems are commonly termed “generalization” and “out-of-library rejection” in the literature. Depending on the context, the consequence of a classification failure can range from inconsequential to catastrophic if the model’s decision is trusted blindly. Laying aside that crucial decisions will likely not be solely entrusted to a computer model, it is nevertheless advantageous to build some measure of robustness and introspection into the machine learning model to aid human decision makers to detect out-of-library targets, and generalization is the de-facto goal of training computer recognition models.

In this paper, we will extend the publicly available Synthetic and Measured Paired Labeled Experiment (SAMPLE) dataset[2*] with data from the Moving and Stationary Target Acquisition and Recognition (MSTAR) dataset[3] and present some experimental designs for investigating the goals of generalization and out-of-library rejection. For naming purposes, we refer to this extension of the SAMPLE dataset as SAMPLE+.

In this paper, we will discuss these two datasets and some of the studies that have been performed on them in Section 2. We will discuss the union of the two datasets in Section 3, including how the data was processed in Subsection 3.1. We will then define a few challenge problems and experiments in Section 4, and present some of our preliminary experiments with this joint dataset in Section 5. Finally, we will discuss our conclusions and avenues for future work in Section 6.

Contact information: {benjamin.lewis.13, mark.ashby.2, edmund.zelnio}@us.af.mil

[*] The public SAMPLE dataset is available on Github at `https://github.com/benjaminlewis-afrl/SAMPLE_dataset_public`. The processed MSTAR data described in this paper is available at `https://github.com/benjaminlewis-afrl/SAMPLE_plus` and should be merged with the SAMPLE dataset, as it contains only a part of the SAMPLE data in the `in_azimuth_in_sample` folder. The authors appreciate feedback and being notified about the dataset’s use.

2. BACKGROUND

The MSTAR dataset is a set of SAR imagery collected by the Defense Advanced Research Projects Agency, Air Force Research Laboratory, and Sandia National Laboratory in 1995 and 1996. The dataset consist of images of ten vehicular targets at X-band with one-foot resolution. Imagery of targets is available at azimuth angles in 360° around each of the vehicles and at depression angles from ≈15-17°. Since its release, this dataset has been the standard SAR image classification dataset and is the subject of hundreds of research studies. Deep learning approaches to the problem can achieve nearly perfect classification accuracy on this dataset when trained at 17° and tested at 15° depression angles.[4–6] However, it has been noted by at least one paper that these models may be at least in part learning the target's background; training on image chips with a black box over the target can still achieve 99% accuracy.[7]

In any case, collecting and curating more airborne SAR imagery is an expensive and time-consuming process, as shown by the MSTAR dataset's enduring popularity nearly three decades later and the still-sparse field of SAR datasets compared to electro-optical image datasets[*]. One reasonable path forward is to instead simulate data using electromagnetic prediction code on computer CAD models. In contrast to data collected with aircraft, simulated data is inexpensive, plentiful, and labeled. These advantages make it worthwhile to investigate methods to train algorithms on synthetic data, despite the differences between synthetic and measured data stemming from approximations made in simulation code and in CAD model design.

To that end, our group released the SAMPLE dataset in 2019.[2] This dataset pairs images of targets from the MSTAR collection with simulated SAR images generated from computer models using asymptotic electromagnetic ray-tracing techniques. For each image in the measured portion of the dataset, a synthetic image of the same target with the same collection geometry was generated. The publicly available portion of this dataset consists of images with azimuth angles between 10 and 80 degrees azimuth.

Since the release of the SAMPLE dataset, the community has performed a wide variety of experiments to investigate ways to bridge the gap between the clean synthetic data and the measured data that is subject to the non-ideal real world, such as insufficient signal-to-noise ratios, backgrounds with non-target objects, and defects and damage to the targets. Among others, approaches have included the use of generative adversarial networks,[8,9] network saliency analysis,[10] adversarial training,[11,12] and investigating hyperparameter optimizations and a variety of image augmentations.[13,14]

For any automated classification algorithm, there are a number of desirable properties, including how reliable the algorithm is, whether it is robust to variability in configuration, how it handles out-of-library confusers, and how it can effectively use synthetic data in its training process. By adding images from MSTAR that are not part of SAMPLE - both of different targets as well as different viewing geometries - we hope to provide the community with data and a challenge problem that can foster research in each of these areas.

3. THE SAMPLE+ DATASET

The SAMPLE+ dataset consists of the data from the SAMPLE dataset plus all publicly released MSTAR data[†]. Although both SAMPLE and MSTAR contain ten targets, the list of targets in the two datasets is different and is shown in Table 1. As seen in this table, there are five targets that are present in both datasets and five targets in each dataset that are unique. In addition, the SLICY object is included in the MSTAR dataset and included here for completeness. SLICY, seen in Figure 1, is a geometric model containing standard radar reflector primitives and was included in the MSTAR collect to allow developers to validate the functionality of their algorithms.

In addition to the targets listed in the table, SAMPLE+ includes the clutter data from the MSTAR collect. While a target classification algorithm will, in theory, be fed pre-filtered data that does not contain non-target image chips, target detection algorithms are subject to false alarms. As such, it is desirable for a classification algorithm to be aware of possible false targets and notify the user of such.

[*] The availability of commercial spaceborne SAR collectors is beginning to change this paradigm for spaceborne SAR, but the challenge of labelling data remains.

[†] Available at https://www.sdms.afrl.af.mil/index.php?collection=mstar, consisting of the MSTAR clutter, MSTAR Target Chips, MSTAR / IU Mixed Targets, and MSTAR T-72 variants products

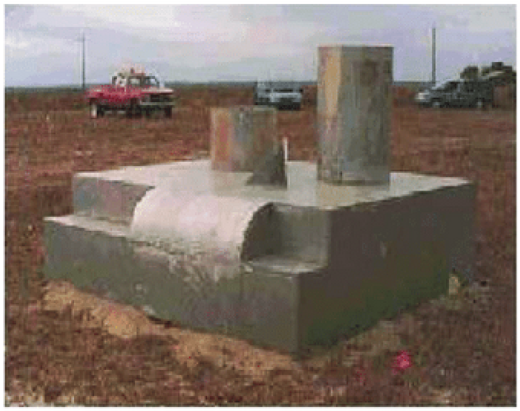


Figure 1: SLICY

	MSTAR Public	SAMPLE
BRDM-2	E-71	None
BTR60	K10YT7532	None
D7	92V13015	None
T62	A51	None
ZIL131	E12	None
M1	None	0AP00N
M2	None	MV02GX
M35	None	T839
M548	None	C245HAB
M60	None	3336
2S1	B01	B01
BTR70	C71	C71
ZSU23-4	D08	D08
BMP2	9563, **c21, 9566**	9563
T72	812, **A63, A04, A62, A32, 132, s7, A64, A10, A05, A07**	812
SLICY *	Included, no serial #	None

Table 1: A list of the targets present in the MSTAR public dataset, SAMPLE public dataset, or in both datasets, along with the available serial numbers. For each target, either the serial numbers present in the dataset are listed, or "None" is listed to indicate that that target is not present in the relevant dataset. MSTAR's unique serial numbers for the BMP2 and T72 are highlighted in bold.

The MSTAR clutter data is available as a set of large-format images taken from the same sensor as the MSTAR images. In general, these images are more than 1000 pixels on a side, as compared to the 128 × 128 image size that is standard in MSTAR (with the exception of SLICY, which is 54 × 54 pixels). To maintain a uniform size, we simply tiled these large images into non-overlapping 128×128 image chips. These clutter images may contain civilian vehicles, structures, and natural clutter that would confuse a classifier. However, we have made no effort to categorize any of these clutter chips or align the image tiling to center on objects in the larger clutter image. We have also processed the full scene images into PNG images so that interested researchers can choose to tile the data differently.

3.1 Data Processing Flow

While the data from MSTAR that is included in the SAMPLE+ dataset is simply the MSTAR data, we have performed a few processing steps beyond simply reading the data and sorting it into appropriate folders. Because machine learning algorithms are sensitive to differences in data format, we deemed it important to apply the same processing steps that were applied to the SAMPLE data, to ensure data homogeneity:

Folder name	Contents
in_azimuth_in_sample	SAMPLE Dataset targets with SAMPLE serial number
in_azimuth_in_sample_high_dep	SAMPLE targets at depression angle > 18
in_azimuth_non_sample	MSTAR-only targets & SAMPLE targets w/non-SAMPLE serial numbers; Clutter
in_azimuth_non_sample_high_dep	Same as above minus clutter at depression angle > 18

Table 2: A list of the in-azimuth folders by name and the set of targets contained in each. There is a corresponding set of "out_azimuth" folders with the same sets of targets but with azimuth angles outside of 10-80°. Since clutter does not have a target azimuth angle by definition, all clutter images are contained in the `in_azimuth_non_sample` folder.

1. Suppress bright outlier pixels using a Grubb's test for outliers. Bright pixels are re-mapped to the brightest value in the non-outlier range.
2. Center crop to 128×128 if necessary (smaller images are not upsized or thrown away)
3. PNG images have either a decibel or quarter-power magnitude normalization applied
4. Matlab-formatted .mat files have no image normalization applied

Regarding image normalization, this process includes 1) normalizing the image to values between 0 and 1, 2) Applying a normalization. For decibel, this is $20 * \log_{10}(x + \epsilon)$ for every pixel ($\epsilon \approx 10^{-16}$ added for numerical stability), while quarter-power magnitude is $\sqrt{x}$ for each pixel, and 3) Re-normalizing the image between integer values of 0 and 255 to map to an 8-bit grayscale PNG image.

For the purposes of reducing the size of the dataset, we do not include any .mat files for the clutter images, as these are created simply by reading the MSTAR data files with the readers provided with MSTAR.

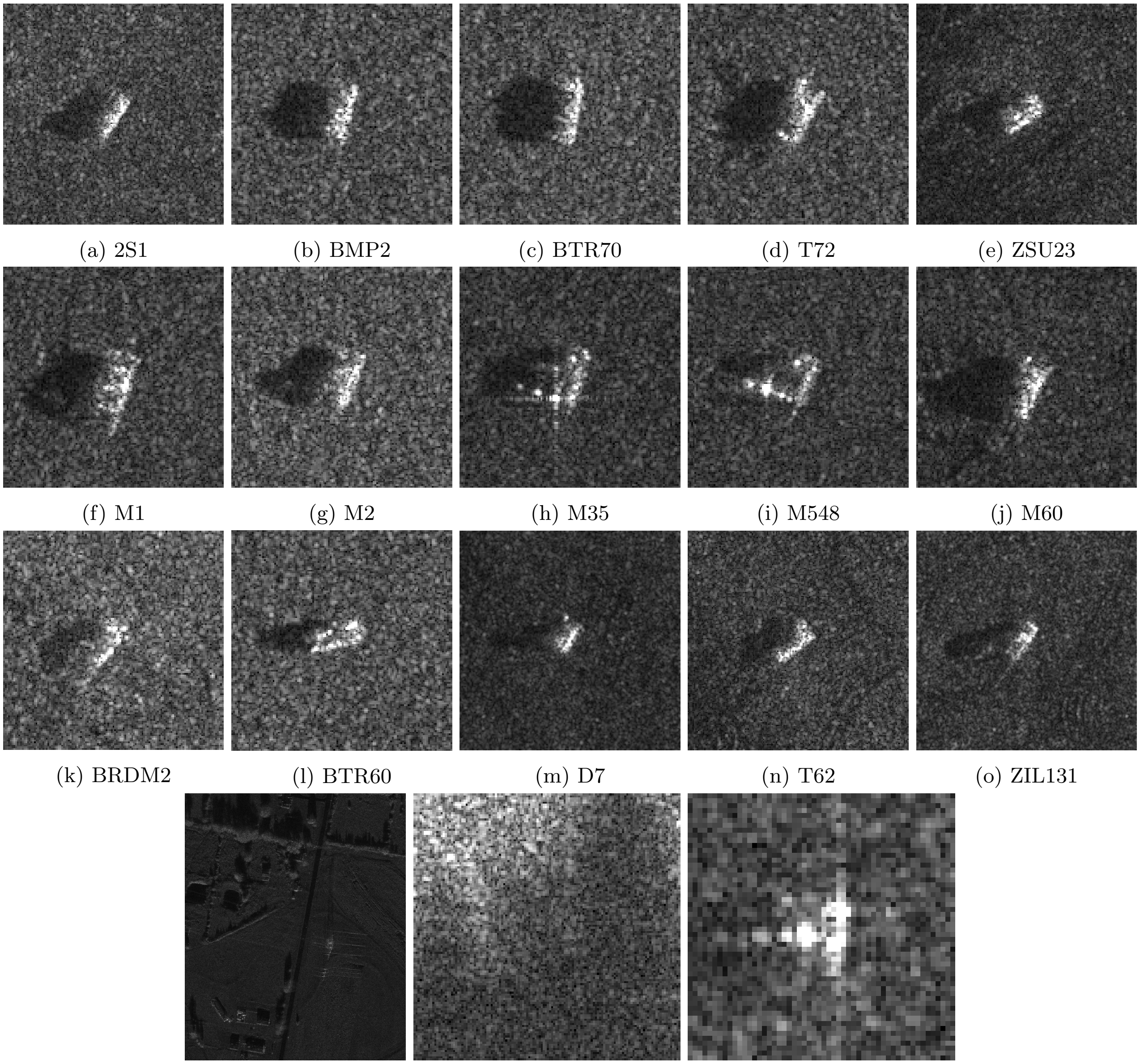
After processing the images into .mat and PNG files, we partitioned the data into folders based on whether the target is part of the SAMPLE dataset or not, whether the azimuth is in the 10-80° azimuth range of SAMPLE or not, and whether the depression angle was in the $\approx 15 - 17°$ covered by SAMPLE. This means that one of these partitions, namely in-azimuth, in-SAMPLE, in-depression angle, is actually composed of the measured portion of SAMPLE. Also note that "part of the SAMPLE dataset" is based on target name as well as serial number, so a T72 with a serial number other than 812 is designated as "out of SAMPLE". A map of target types to folder name for in-azimuth targets is shown in Table 2. The corresponding out-of-azimuth folders contain the same target sets but with target azimuth values outside of the 10-80° range. Because clutter does not contain identified targets with a defined azimuth value, all clutter is contained in the `in_azimuth_non_sample` folder.

There are a few other minor differences between SAMPLE and SAMPLE+, mostly at a surface level:

- The file naming convention is slightly different; we included thousandths of a degree in the elevation portion of the name here, but not in the original SAMPLE dataset.
- We included images (SLICY) that are smaller than the 128×128 pixels of SAMPLE. They are tagged with an appended "_undersized" label. Other images were downsampled where appropriate with a simple center crop.
- Unlike with SAMPLE, no effort was made to center targets exactly in the frame. In SAMPLE, this information was saved as `offset_information` in .mat files.
- Some targets, namely BMP2 and T72, have more serial numbers than were represented in the SAMPLE dataset. This means that serial number is relevant for some of the experiments outlined in this paper.
- As mentioned earlier, we include the clutter images in tiled form.
- A couple of miscellaneous metadata fields - range and cross-range extent - were removed from the .mat files.

- Each image has its source image in the MSTAR dataset identified in the "filepath" field.

Example images of each of the various classes, both from MSTAR and from SAMPLE, are shown in Figure 2.



(a) 2S1 (b) BMP2 (c) BTR70 (d) T72 (e) ZSU23

(f) M1 (g) M2 (h) M35 (i) M548 (j) M60

(k) BRDM2 (l) BTR60 (m) D7 (n) T62 (o) ZIL131

(p) Full Clutter Scene (q) Clutter Chip (r) SLICY

Figure 2: Example chips from the dataset, including vehicles in both MSTAR and SAMPLE (a-e), SAMPLE only (f-j), MSTAR only (k-o), and clutter chips and SLICY (p-r).

4. CHALLENGE PROBLEMS

The addition of out-of-library targets, in-library targets in different configurations, additional azimuth angles, and clutter data open several interesting possibilities for extending work on the SAMPLE challenge problem and training algorithms with synthetic data. In order to foster work in this area, we will describe a few challenge problems that can be undertaken with the SAMPLE+ dataset. In each subsection, we will describe a number of experiments that can be conducted with this dataset, as well as one “canonical” experiment that serves as an interesting starting point for the problem and that can be referenced for easy comparison. However, the canonical experiment suggested here does not preclude other experimentation.

4.1 Accuracy across serial numbers

Machine learning algorithms are capable of building powerful representations, but for good generalization performance, the training data must be representative of the testing or operational data. While vehicles in a given class are broadly similar, there can be differences between instances of the vehicle, represented by serial number in this dataset. These differences can be accounted for in three main categories.

First, the individual vehicle configuration may vary between serial numbers. By analogy, two pickup trucks may come off the assembly line with the same configuration, but one owner may decide to mount a tool chest in the bed of the truck and the other get a truck bed cover. This would change the radar characteristics of individual serial numbers, even though they are the same model of truck. With military vehicles, similar changes may occur based on unit needs.

Second, different vehicles undergo different service lives and will inevitably have dissimilar patterns of wear and dents, causing subtle differences in the radar return. Additionally, even at the manufacturing stage, it is conceivable that changes in tooling at the plant or changes in materials may slightly vary the geometry or radar properties of the target.

Third, individual vehicles are likely to have a different configuration during a collect and a different clutter background. For example, the gun turret may be oriented facing forward on one vehicle and sideways on a different instance of the same target.

The total effect of these three variations can be significant and lead to poor performance on a given target with a different serial number. An experiment in this domain could involve training a classifier with the SAMPLE data - either solely with synthetic or with measured and synthetic - and then testing classifier performance on the other serial numbers for the BMP2 and T72. Accuracy may be reported on a per-serial-number basis as well as cumulatively. An easy test would be the two-target classifier, while a more realistic test would involve the ten SAMPLE targets or the 15 targets included in the union of SAMPLE and MSTAR.

The canonical experiment for this approach involves training a classifier on the 10 targets in the SAMPLE synthetic data, then classifying all of the measured data for the SAMPLE measured data as well as the in-azimuth non-SAMPLE serial numbers for the BMP2 and T72. Accuracy is reported on a per-serial-number basis, with individual accuracies for each serial number in BMP2 and T72 but aggregate classifier accuracy weighted by class (e.g. T72 accuracy for all serial numbers is weighted the same as 2S1 accuracy).

4.2 Rejecting out-of-domain targets

It is often important to identify targets that are not in the set of targets the network was trained on. This has been a topic of research in the machine learning community recently, with techniques such as ODIN,[15] energy-based out-of-distribution detection,[16] ensembling techniques,[17] and distribution calibration[18] applied in recent years.

The SAMPLE+ dataset has a good structure for this type of experiment. While many experiments are possible, there is one experiment for this category that is fairly obvious. In this experiment, the classifier is trained on the synthetic SAMPLE data, possibly with a small percentage of measured data, and tested on the measured SAMPLE and MSTAR data. The network is tasked with classifying the SAMPLE targets while correctly rejecting/identifying the targets - BRDM-2, BTR60, D7, T62, ZIL131 - that are only available in the MSTAR data and that should not be classified as one of the SAMPLE targets. Metrics for this experiment include the precision and recall rate for identifying whether a target is a member of the SAMPLE targets and classification accuracy on SAMPLE targets that are identified as in-domain.

4.3 Rejecting target clutter

This application is substantially similar to rejecting out-of-domain targets, although it could arguably be seen as a slightly easier task because the clutter chips do not have the vehicle-like structure seen in the out-of-domain targets. As mentioned earlier, an ideal target detection algorithm would never feed such false alarms to the classifier, this simply cannot always be the case, and it is advantageous for the classifier to be able to reject clutter that is passed to the network. Much of the clutter present in the MSTAR collect consists of backgrounds of fields or forested areas, but there are some scenes that include buildings and vehicles which can be confusers.

The canonical experiment for this problem is again to test the ability of a classifier to correctly identify image chips that do not contain a target of interest. The network can be trained on the synthetic SAMPLE data, possibly with a small percentage of measured data, and is tested against the remaining measured SAMPLE data and the clutter data. The classifier is then measured on the precision and recall for identifying whether an image chip is a SAMPLE dataset target or a clutter chip.

4.4 Elevation diversity

There are a few targets that are available at high depression angles - 30° or 45° - 2S1 and ZSU23-4 for SAMPLE targets and BRDM2 and T72 for non-SAMPLE targets/serial numbers. Because these elevations lack the full set of MSTAR targets, they aren't amenable to a full experiment like the full set of targets would be. However, a possible experiment with this involves training on the SAMPLE and/or MSTAR data, possibly projected into the ground plane, and then either classifying these targets correctly or detecting that they are out-of-distribution targets or at too high of an elevation.

4.5 Withheld azimuths

A final experiment involves the azimuth angles that are out of the 10-80° wedge. When the SAMPLE dataset was released, it was decided to only publicly release this range of azimuths. By supplementing with the measured images from the remainder of the azimuth range, researchers can perform a number of experiments regarding network confidence, out-of-domain rejection, and identifying commonalities between the in-azimuth and out-of-azimuth data.

We suggest a canonical experiment with these azimuths that explores how classification accuracy varies as a function of azimuth value. The network is trained on the SAMPLE synthetic data, then evaluated against all measured SAMPLE data and the measured data for the targets that are present in both SAMPLE and MSTAR. Accuracy can be reported for 15-degree bins in azimuth. We would expect to see good accuracy in the SAMPLE azimuth angles of 10-80°, and perhaps reasonably good accuracy for 280-350°, which are to first order a view of the target reflected across the 0° axis. It may be advantageous to rotate targets to a standard heading or investigate other methods to make the images look consistent regardless of rotation angle.

5. EXPERIMENTAL RESULTS

We explore variations of the challenge problems described above to demonstrate the utility of this dataset and encourage further related research. We focus our experimentation on the challenge problems described in Section 4.1 and Section 4.2, i.e. generalizing to different serial numbers and rejecting out-of-library confusers.

5.1 Experimental Approach

Our experimentation incorporates the Adversarial Reciprocal Points Learning (ARPL) open set recognition algorithm described in Chen et al.[19] This algorithm optimizes a CNN such that the empirical classification risk and the open space risk are jointly minimized. In other words, the algorithm learns to discriminate in-domain targets while still maintaining the ability to recognize out-of-domain targets. The loss function includes a margin constraint term which bounds the open space and forces out-of-domain features to inhabit the feature space closer to the origin, which is natural due to their low activation. This margin constraint is enforced by two learned parameters: an overall radius in the feature space and the distance between the "reciprocal point" of each class and the mean of its features. Reciprocal points are points in feature space that represent features that are adverse to the given class - e.g., for a T72, the reciprocal point represents image features that are "non-T72".

A cross entropy loss term is included in the cost function to drive separation between the in-domain targets toward the periphery of the bounded feature space away from the origin. ARPL also uses a generative adversarial network (GAN) to create confusing samples, which are meant to appear similar to in-domain targets but produce low entropy and thus reside in the feature space near the origin where out-of-domain targets are expected to reside. The confusing samples are used during training to reduce the average magnitude of out-of-domain features and thus increase the separation between the in-domain and out-of-domain targets in order to yield better open set recognition performance.

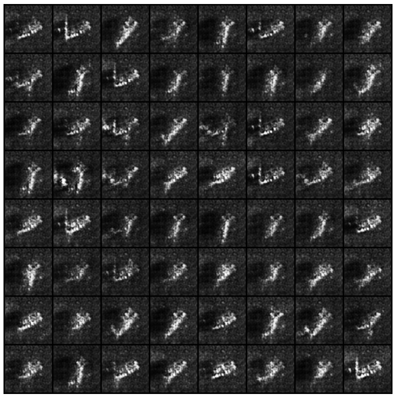


Figure 3: GAN-generated confusing samples after 40 epochs of training with SAMPLE+

Datasets like SAMPLE+ support algorithm development in a transfer learning setting in which most if not all training data is synthetic. Often deep learning classifiers have low classification and recognition performance on these difficult problems and yet remain highly confident in their predictions. Thus, we also incorporate a network calibration step after training to adjust prediction probabilities given the presence of a small amount of measured data. Very few parameters need to be learned during calibration compared to the number learned during CNN training. Temperature scaling, vector scaling, and matrix scaling only require 1, k and k^2 parameters respectively, where k is the number of target classes.[20] With so few parameters, very little data is needed. Thus, calibration has utility to quickly update a fully-trained model when encountering new data in the wild.

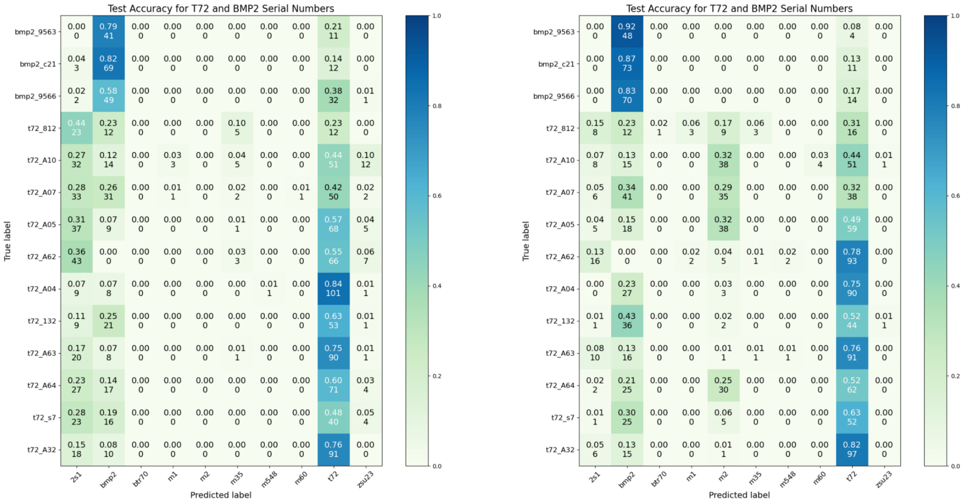
For calibration to be useful on open set problems, it is necessary to have a probability vector that includes confusers. ARPL generates predictions with a scoring mechanism that is based on the euclidean distance and dot product between the sample's feature vector and the reciprocal point of each class. A greater value for the score indicates that the sample is near the center point for that class and should be classified as such.

ARPL does not natively generate a probability for its confuser class since targets that are not in-domain do not have a reciprocal point. We modified the ARPL algorithm to generate a surrogate score for the confuser class $S_c = R - |x|_2$ where R is the learnable parameter restricting the open space and x is the feature vector of the sample. S_c is maximized when the magnitude of the feature vector is zero. We have demonstrated success using this score in conjunction with the scores already used for in-domain targets during training and testing. Since confusers are not seen until testing, we apply cross entropy loss for the GAN-generated confusing samples during training and thus jointly optimize the network to predict in-domain and out-of-domain targets.

With scores for all targets, it is possible to calibrate predictions and in fact use this calibration data for statistically rigorous conformal prediction to generate prediction sets, which contain the true target with some degree of confidence.[21] However, conformal prediction will not be included in the following experiments.

5.2 Experimental Setup

All of the following results were generated using a LeNet++ CNN architecture.[22] While a more sophisticated network would likely perform better, we expect our approach to generalize across all network types. All classification results are from 30 epochs of training with Adam optimization and a learning rate of 0.0001. The GAN learning rate was 0.0002. The weight for the confusing sample entropy loss was 0.05 and the weight for the RPL margin loss was 0.1. When performing confuser rejection, the radius parameter was initialized at 3.0. Results



	2s1	bmp2	btr70	m1	m2	m35	m548	m60	t72	zsu23
bmp2_9563	0.00 0	0.79 41	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.21 11	0.00 0
bmp2_c21	0.04 3	0.82 69	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.14 12	0.00 0
bmp2_9566	0.02 2	0.58 49	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.38 32	0.01 1
t72_812	0.44 23	0.23 12	0.00 0	0.00 0	0.00 0	0.10 5	0.00 0	0.00 0	0.23 12	0.00 0
t72_A10	0.27 32	0.12 14	0.00 0	0.03 3	0.00 0	0.04 5	0.00 0	0.00 0	0.44 51	0.10 12
t72_A07	0.28 33	0.26 31	0.00 0	0.01 1	0.00 0	0.02 2	0.00 0	0.01 1	0.42 50	0.02 2
t72_A05	0.31 37	0.07 9	0.00 0	0.00 0	0.00 0	0.01 1	0.00 0	0.00 0	0.57 68	0.04 5
t72_A62	0.36 43	0.00 0	0.00 0	0.00 0	0.00 0	0.03 3	0.00 0	0.00 0	0.55 66	0.06 7
t72_A04	0.07 9	0.07 8	0.00 0	0.00 0	0.00 0	0.00 0	0.01 1	0.00 0	0.84 101	0.01 1
t72_132	0.11 9	0.25 21	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.63 53	0.01 1
t72_A63	0.17 20	0.07 8	0.00 0	0.00 0	0.00 0	0.01 1	0.00 0	0.00 0	0.75 90	0.01 1
t72_A64	0.23 27	0.14 17	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.60 71	0.03 4
t72_s7	0.28 23	0.19 16	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.48 40	0.05 4
t72_A32	0.15 18	0.08 10	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.76 91	0.00 0

	2s1	bmp2	btr70	m1	m2	m35	m548	m60	t72	zsu23
bmp2_9563	0.00 0	0.92 48	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.08 4	0.00 0
bmp2_c21	0.00 0	0.87 73	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.13 11	0.00 0
bmp2_9566	0.00 0	0.83 70	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.17 14	0.00 0
t72_812	0.15 8	0.23 12	0.02 1	0.06 3	0.17 9	0.06 3	0.00 0	0.00 0	0.31 16	0.00 0
t72_A10	0.07 8	0.13 15	0.00 0	0.00 0	0.32 38	0.00 0	0.00 0	0.03 4	0.44 51	0.01 1
t72_A07	0.05 6	0.34 41	0.00 0	0.00 0	0.29 35	0.00 0	0.00 0	0.00 0	0.32 38	0.00 0
t72_A05	0.04 5	0.15 18	0.00 0	0.00 0	0.32 38	0.00 0	0.00 0	0.00 0	0.49 59	0.00 0
t72_A62	0.13 16	0.00 0	0.00 0	0.02 2	0.04 5	0.01 1	0.02 2	0.00 0	0.78 93	0.00 0
t72_A04	0.00 0	0.23 27	0.00 0	0.00 0	0.03 3	0.00 0	0.00 0	0.00 0	0.75 90	0.00 0
t72_132	0.01 1	0.43 36	0.00 0	0.00 0	0.02 2	0.00 0	0.00 0	0.00 0	0.52 44	0.01 1
t72_A63	0.08 10	0.13 16	0.00 0	0.00 0	0.01 1	0.01 1	0.01 1	0.00 0	0.76 91	0.00 0
t72_A64	0.02 2	0.21 25	0.00 0	0.00 0	0.25 30	0.00 0	0.00 0	0.00 0	0.52 62	0.00 0
t72_s7	0.01 1	0.30 25	0.00 0	0.00 0	0.06 5	0.00 0	0.00 0	0.00 0	0.63 52	0.00 0
t72_A32	0.05 6	0.13 15	0.00 0	0.00 0	0.01 1	0.00 0	0.00 0	0.00 0	0.82 97	0.00 0

Figure 4: An example of BMP2 and T72 serial number accuracy before and after vector scaling with 32 samples per class. Pre-calibration results are shown on the left. Results after vector scaling are shown on the right.

show the average of five runs with different random seeds. The analysis was done with quarter-power magnitude data, not decibel.

5.3 Experiment 1: Accuracy Across Serial Numbers

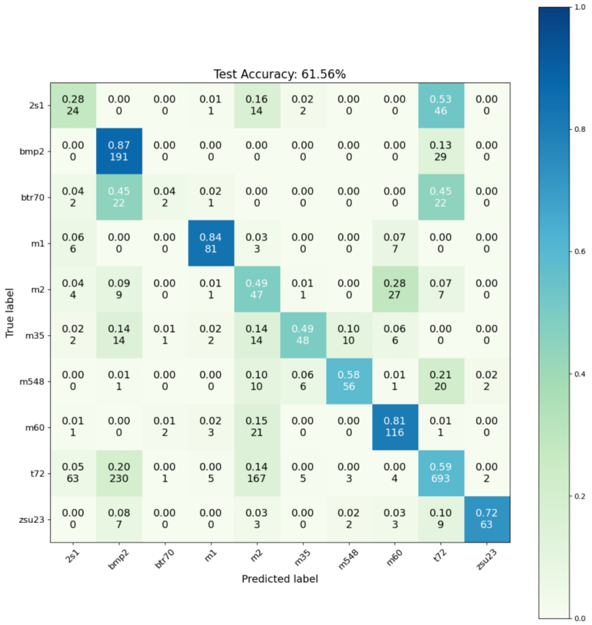
This experiment follows the approach outlined in Section 4.1 in which we test a network's ability to accurately classify targets with serial numbers that are not present in the training dataset.

It is a ten-class problem where the training set consists of all synthetic SAMPLE data. After training, the network is calibrated on a small number of images from the SAMPLE in-azimuth measured data as well as extra MSTAR out-azimuth data for the BMP2 and T72. Our tests steadily increased the number of calibration samples per class by a power of two: 2, 4, 8, 16, 32, and 64.

The test set for this experiment consists of all measured SAMPLE data not used during calibration in addition to the MSTAR in-azimuth data for the different BMP2 and T72 serial numbers. Since the network never encounters those serial numbers during training, the aim is for the network to generalize and correctly classify these targets using a small amount of measured data at a different azimuth wedge during calibration, as well as to correctly classify the eight other test targets. While the ARPL algorithm is in principle designed for confuser rejection, it can also be used for closed-set problems with out-of-distribution test samples such as this, hence our choice to use it here.

In the left portion of Figure 4, we show the confusion matrix from one run using no calibration data. The true serial numbers are on the y-axis and the predicted target is on the x-axis. Classification performance on the different serial numbers varies greatly, reaching as high as 84 percent and as low as 23 percent. The 2S1 is the main confusing class for the T72 while the T72 is the main confusing class for the BMP2.

The confusion matrix on the right side of Figure 4 shows the results from the same run after calibrating with the vector scaling method. In this case, the classifier was given 32 measured samples from each of the in-domain targets as well as 32 samples for the out-of-domain serial numbers in the MSTAR out-azimuth dataset. This new classifier applies a single weight to the probabilities of each class, which results in predictions that are different from the uncalibrated predictions. Accuracy improves for many of the serial numbers, including those that did not appear in the training data. Overall accuracy for the same test case across all classes is shown in the vector-scaled confusion matrix in Figure 5. The BMP2 had 87 percent overall accuracy while the T72, which has more serial number variation and thus might be considered a harder target to classify, had just 59 percent. Overall accuracy in this case was 62 percent. The improvement on serial numbers that did not appear in the training data indicates that calibrating with out-azimuth data provides additional predictive power.



	2s1	bmp2	btr70	m1	m2	m35	m548	m60	t72	zsu23
2s1	0.28 24	0.00 0	0.00 0	0.01 1	0.16 14	0.02 2	0.00 0	0.00 0	0.53 46	0.00 0
bmp2	0.00 0	0.87 191	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.00 0	0.13 29	0.00 0
btr70	0.04 2	0.45 22	0.04 2	0.02 1	0.00 0	0.00 0	0.00 0	0.00 0	0.45 22	0.00 0
m1	0.06 6	0.00 0	0.00 0	0.84 81	0.03 3	0.00 0	0.00 0	0.07 7	0.00 0	0.00 0
m2	0.04 4	0.09 9	0.00 0	0.01 1	0.49 47	0.01 1	0.00 0	0.28 27	0.07 7	0.00 0
m35	0.02 2	0.14 14	0.01 1	0.02 2	0.14 14	0.49 48	0.10 10	0.06 6	0.00 0	0.00 0
m548	0.00 0	0.01 1	0.00 0	0.00 0	0.10 10	0.06 6	0.58 56	0.01 1	0.21 20	0.02 2
m60	0.01 1	0.00 0	0.01 2	0.02 3	0.15 21	0.00 0	0.00 0	0.81 116	0.01 1	0.00 0
t72	0.05 63	0.20 230	0.00 1	0.00 5	0.14 167	0.00 5	0.00 3	0.00 4	0.59 693	0.00 2
zsu23	0.00 0	0.08 7	0.00 0	0.00 0	0.03 3	0.00 0	0.02 2	0.03 3	0.10 9	0.72 63

Figure 5: Accuracy across all classes when vector scaling with 32 calibration samples per class is applied.

We tested this hypothesis with further experimentation over a series of five training iterations and six different calibration sample sizes. Figure 6 shows the summary statistics from those experiments in terms of overall accuracy, T72 accuracy, and BMP2 accuracy. The bands around each of the lines in the graphs represent a 90 percent bootstrapped confidence interval. It is clear from the figure that vector scaling performs very well relative to no calibration. There is even a measurable performance boost when only two calibration samples per class are provided. As the amount of calibration data increases, there are diminishing gains.

Matrix scaling shows some improvement over no calibration, but falls short compared to vector scaling. This is an interesting result since vector scaling requires the classifier to learn just k parameters compared to the k^2 parameters needed for matrix scaling. With as few as eight samples per class, vector scaling offers more than a ten percent gain in overall classification accuracy and costs very little computationally.

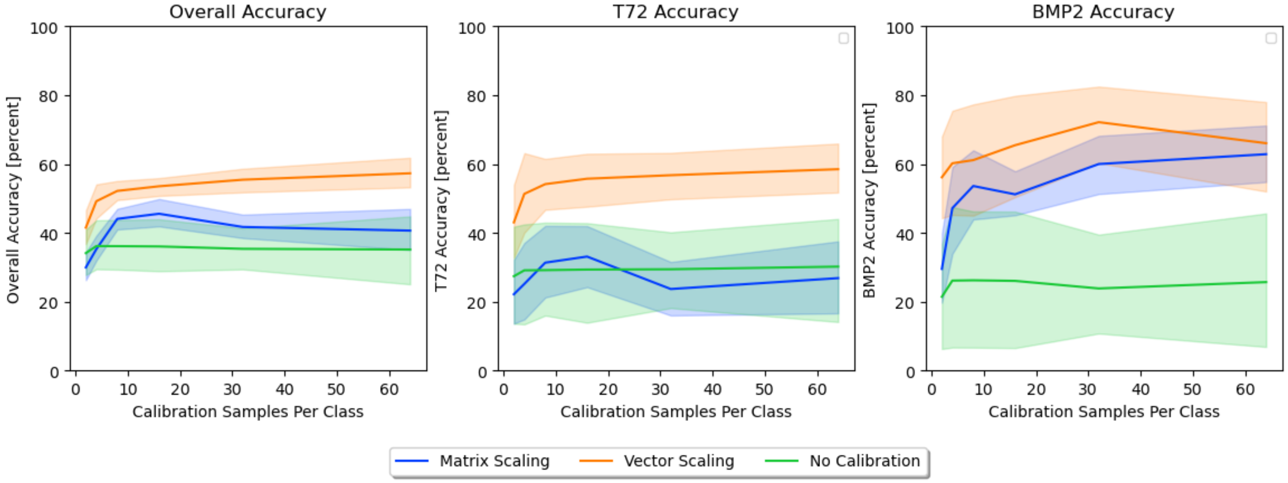


Figure 6: Average classification accuracy with 90 percent CI bands with and without calibration.

The expected calibration error (ECE) is a metric that quantifies how well the classifier is calibrated, i.e.

how well its accuracy aligns with its confidence. ECE divides the predictions into equally spaced bins of size n and takes a weighted average of the difference between accuracy and confidence across all bins.[20] A perfectly calibrated network would be one where ECE is zero.

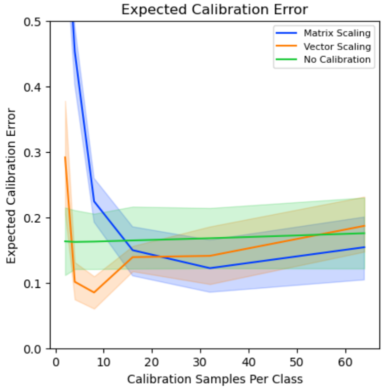


Figure 7: ECE with and without calibration.

The ECE values for vector scaling, matrix scaling, and no calibration are shown in Figure 7. The vector-scaled and matrix-scaled probabilities generally have lower ECE values than the uncalibrated probabilities except when the number of calibration samples is extremely low. However, ECE is still relatively high compared to well-calibrated networks. The overconfidence of our network suggests that using out-azimuth measured data for the serial numbers not present in training improves performance but still may be insufficient to fully align accuracy with confidence.

5.4 Experiment 2: Rejecting Out-of-Domain Targets

This experiment focuses on the fifteen-class open-set problem described in Section 4.2 where the ten SAMPLE targets are used in testing and the five in-azimuth MSTAR targets which are not present in SAMPLE are used as out-of-library confusers. The aim of this experiment is to train a classifier that is able to recognize the presence of unknown targets while simultaneously maintaining high classification accuracy among the known targets. Once again, we use all synthetic data for training and a small amount of measured data for calibration. The in-azimuth SAMPLE measured data is used to calibrate the ten in-domain targets. However, the out-of-domain confuser class is not calibrated on in-azimuth data since strictly speaking this would no longer make the class out-of-domain.

Instead, we try calibrating the confuser class in three different ways:

1. **Use all available SLICEY data for calibration.** 186 samples of in-azimuth SLICEY data is provided in the SAMPLE+ dataset. While SLICEY (pictured in Figure 1) is not a ground vehicle, it shares some similar geometry to SAMPLE+ targets which might be sufficient to calibrate a confuser class. At the very least, SLICEY provides more measured data, which the classifier so far has not seen. Since SLICEY data is readily available, we used all of it for calibration even when the in-domain targets were restricted to fewer calibration samples.

2. **Use GAN-generated confusing samples for calibration.** The purpose of the GAN is to produce data that is highly similar to the in-domain targets while residing away from them in the feature space. We can continue this process into calibration with a fresh batch of 64 GAN samples. Like with SLICEY, we did not vary the number of GAN samples even as the calibration data for in-domain targets varied since the GAN samples are "free" in the sense that they are not measured data that would need to be collected.

3. **Use out-azimuth MSTAR data from each confuser class for calibration.** Unlike with SLICEY and the GAN, in this case the in-domain and out-of-domain targets always have an equal number of calibration samples.

These three options are quite distinct and therefore interesting to compare. SLICEY is akin to an “in-library” measured confuser, the GAN is a purely synthetic data solution, and the out-azimuth data represents the scenario where target data exists but is highly limited.

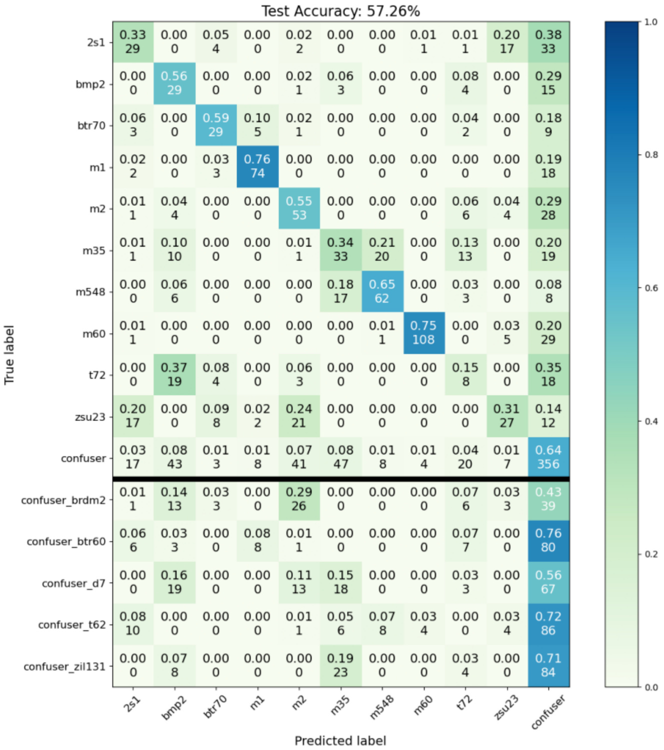


Figure 8: Example of accuracy in open set problem with 32 samples per class of calibration data used to perform matrix scaling. The confuser calibration data is from out-azimuth MSTAR targets.

To provide a sense of performance across classes, we show a confusion matrix for one run of the ARPL-trained CNN in Figure 8. This example is from the third case in which the confuser class is calibrated on the out-azimuth MSTAR data using the matrix scaling method. Note that the classifier makes predictions for a single confuser class and does not distinguish between the different confuser vehicle types. However, the confusion matrix shows the specific vehicle types in a breakout below the main part of the figure. After matrix scaling, the classifier had 64 percent precision and 65 percent recall for the confuser class while maintaining an overall accuracy in this experiment of 57 percent. Of the confuser vehicle classes, the BRDM2 was the most stressing case in terms of precision. While Figure 8 shows results from a single example, structured experimentation follows.

As in Section 5.3, this analysis includes the results of five training iterations and six different calibration sample sizes. Figure 9 compares overall accuracy, confuser class accuracy, and ECE for the three confuser calibration methods when matrix scaling is applied. In terms of all three metrics, the SLICEY-calibrated network performs the worst. Its overall accuracy hovers near 45 percent at its peak while confuser class accuracy drops to below 20 percent, and ECE flattens out at around 0.3. The fact that SLICEY is relatively ineffective for confuser calibration is perhaps not surprising since it visually looks quite dissimilar to other targets in the SAMPLE+ dataset. However, SLICEY is a good baseline to compare to the other two methods.

Calibrating the confuser class on the GAN-generated confusing samples leads to an improvement in all areas over SLICEY calibration. The accuracy of the classifier on the confuser class is the highest of all three methods. This is most likely because the classifier has been trained to recognize similar GAN samples during training and therefore classifies them quite accurately during calibration. A high classification rate during calibration suggests that the classifier will be confident in its predictions of the confuser class and predicts that class often as a result.

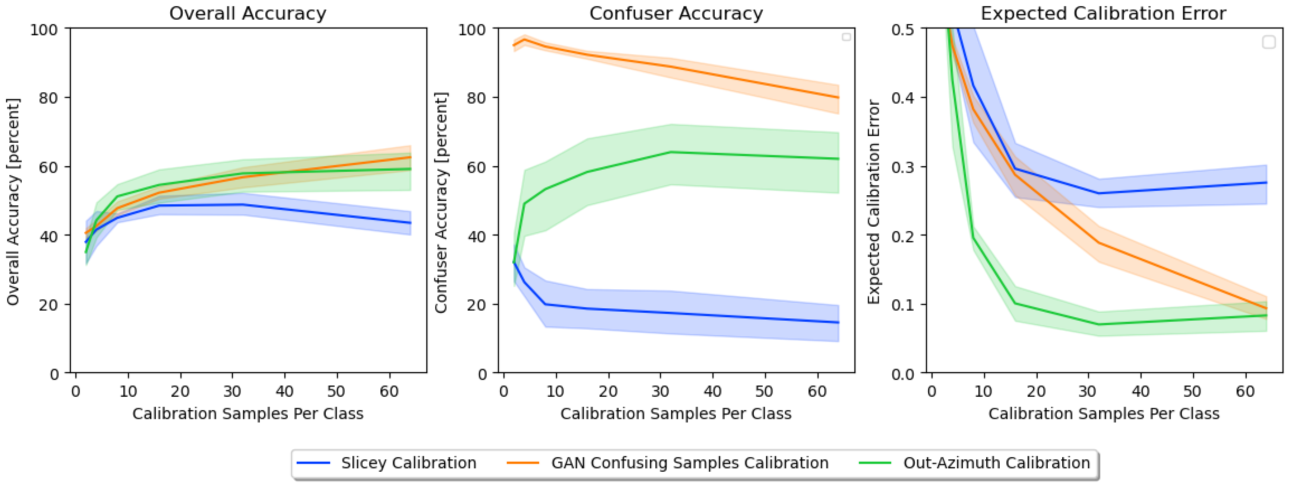


Figure 9: Accuracy and ECE in open set problem when matrix scaling is applied. Figures show differences in performance when SLICEY, GAN confusing samples, and out-azimuth MSTAR targets are used to calibrate the confuser class.

Of all three methods, out-azimuth calibration seems most promising. While it classifies confusers more poorly than the GAN calibration, it has slightly higher overall accuracy in most cases and a much lower ECE score. This is not a terribly surprising result since this method uses measured data from the true confuser targets. In the future, it would be interesting to try a combination of these ideas. Furthermore, it would be useful to try calibrating the confuser class on other "in-library" confusers that are more realistic than SLICEY.

Lastly, instead of calibrating with samples from a GAN that is generated on purely synthetic data, it would be interesting to try calibrating on samples from a GAN trained on some measured data from the in-domain targets. This would provide highly realistic fake targets that are only slightly different from the in-domain targets. We expect that calibrating the confuser class on these targets would be a difficult task for the classifer and therefore highly effective at lowering calibration error and increasing the network's ability to reject confusers during testing.

6. CONCLUSIONS

In this paper, we have introduced the SAMPLE+ dataset, a series of challenge problems that can be addressed with this dataset, and some initial experimentation that follow the structure of two of those challenge problems. Outlier detection is an important subfield of machine learning, especially in applications with high demands for accuracy. By augmenting the SAMPLE dataset with the public MSTAR data, we hope that the community will continue to make progress in outlier rejection and generalization for SAR imagery.

Distribution A: Approved for public release. Distribution unlimited. PA Approval APRS-RYA-2023-04-00002.

REFERENCES

[1] Krizhevsky, A., Sutskever, I., and Hinton, G. E., "Imagenet classification with deep convolutional neural networks," in [*Advances in neural information processing systems*], 1097–1105 (2012).

[2] Lewis, B., Scarnati, T., Sudkamp, E., Nehrbass, J., Rosencrantz, S., and Zelnio, E., "A SAR Dataset for ATR Development: the Synthetic and Measured Paired Labeled Experiment (SAMPLE)," in [*SPIE Algorithms for Synthetic Aperture Radar*], (2019).

[3] Sandia National Laboratory, "MSTAR Overview." https://www.sdms.afrl.af.mil/index.php?collection=mstar (1995).

[4] Blasch, E., Majumder, U., Zelnio, E., and Velten, V., "Review of recent advances in AI/ML using the MSTAR data," in [*Algorithms for Synthetic Aperture Radar Imagery XXVII*], **11393**, 113930C, International Society for Optics and Photonics (2020).

[5] Chen, S. and Wang, H., "SAR target recognition based on deep learning," *2014 International Conference on Data Science and Advanced Analytics (DSAA)* (2014).

[6] Chen, S., Wang, H., Xu, F., and Jin, Y.-Q., "Target Classification Using the Deep Convolutional Networks for SAR Images," *IEEE Transactions on Geoscience and Remote Sensing* (2016).

[7] Schumacher, R. and Schiller, J., "Non-cooperative target identification of battlefield targets - Classification results based on SAR images," in [*IEEE National Radar Conference - Proceedings*], **2005-Janua**(January), 167–172 (2005).

[8] Lewis, B., Liu, J., and Wong, A., "Generative Adversarial Networks for SAR Image Realism," in [*SPIE Algorithms for Synthetic Aperture Radar*], (2018).

[9] Lewis, B., DeGuchy, O., Sebastian, J., and Kaminski, J., "Realistic SAR Data Augmentation using Machine Learning Techniques," in [*SPIE Algorithms for Synthetic Aperture Radar*], (2019).

[10] Lewis, B., Scarnati, T., and Parke, E., "Investigating the saliency of SAR image chips," in [*SPIE Algorithms for Synthetic Aperture Radar*], (2020).

[11] Inkawhich, N., Inkawhich, M. J., Davis, E. K., Majumder, U. K., Tripp, E., Capraro, C., and Chen, Y., "Bridging a Gap in SAR-ATR: Training on Fully Synthetic and Testing on Measured Data," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **14**, 2942–2955 (2021).

[12] Lewis, B., Cai, K., and Bullard, C., "Adversarial Training on SAR Images," in [*SPIE Algorithms for Synthetic Aperture Radar*], (2020).

[13] Melzer, R., Severa, W. M., Plagge, M., and Vineyard, C. M., "Exploring characteristics of neural network architecture computation for enabling SAR ATR," **11729**, 7, SPIE Algorithms for Synthetic Aperture Radar (2021).

[14] Melzer, R., Severa, W. M., and Vineyard, C. M., "Exploring SAR ATR with neural networks: going beyond accuracy," (May), 14 (2022).

[15] Liang, S., Li, Y., and Srikant, R., "Principled detection of out-of-distribution examples in neural networks," *CoRR* **abs/1706.02690** (2017).

[16] Liu, W., Wang, X., Owens, J., and Li, Y., "Energy-based out-of-distribution detection," *Advances in neural information processing systems* **33**, 21464–21475 (2020).

[17] Vyas, A., Jammalamadaka, N., Zhu, X., Das, D., Kaul, B., and Willke, T. L., "Out-of-distribution detection using an ensemble of self supervised leave-out classifiers," in [*Proceedings of the European Conference on Computer Vision (ECCV)*], 550–564 (2018).

[18] Wu, Y., Zeng, Z., He, K., Mou, Y., Wang, P., and Xu, W., "Distribution calibration for out-of-domain detection with bayesian approximation," *arXiv preprint arXiv:2209.06612* (2022).

[19] Chen, G., Peng, P., Wang, X., and Tian, Y., "Adversarial reciprocal points learning for open set recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence* , 1–1 (2021).

[20] Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q., "On calibration of modern neural networks," (2017).

[21] Angelopoulos, A. N. and Bates, S., "A gentle introduction to conformal prediction and distribution-free uncertainty quantification," (2022).

[22] LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P., "Gradient-based learning applied to document recognition," *Proceedings of the IEEE* **86**(11), 2278–2324 (1998).