Improving Automatic Target Recognition in Low Data Regime using Semi-Supervised Learning and Generative Data Augmentation

Anonymous CVPR submission

Paper ID *****

Abstract

We propose a new strategy to improve Automatic Target Recognition (ATR) from infrared (IR) images by leveraging semi-supervised learning and generative data augmentation. Our approach is twofold: first, we use an automatic detector's outputs to augment the existing labeled and unlabeled data. Second, we introduce a confidence-guided data generative augmentation technique that focuses on learning from the most challenging regions of the feature space, to generate synthetic data which can be used as extra unlabeled data.

The proposed approach yields substantial percentage improvements in ATR performance on the DSIAC dataset relative to both the baseline fully supervised model trained using the existing data only, and a semi-supervised model trained without generative data augmentation. For instance, for the most challenging data partition, our method achieves a relative increase of 29.51% over the baseline fully supervised model and a relative improvement of 2.59% over the semi-supervised model. These results demonstrate the effectiveness of our approach in low-data regimes, where labeled data is limited or expensive to obtain.

1. Introduction

Automatic Target Recognition (ATR) from infrared images is an important task in computer vision with many practical applications in security, emergency services, automotive, environment, and other fields [8].

Various deep learning-based methods have been proposed for ATR from RGB images. Howerver, they do not perform as well in the infrared domain, where the lack of color information and other environmental factors make the task more challenging [11, 24]. Recent works have attempted to address this problem by leveraging deep learning techniques for feature extraction, transfer learning, and data augmentation [10]. Nevertheless, these approaches require a large amount of labeled data, which is often hard to obtain in infrared ATR due to the high cost of collecting and labeling data. In fact, infrared sensors provide valuable information for ATR, especially in low-light conditions, but still pose unique challenges due to the difficulty in acquiring high-resolution labeled inputs, and sensitivity to calibration and environmental conditions. These challenges limit the effectiveness of fully supervised ATR models, which require large amounts of labeled data that are tedious to obtain [14].

Semi-supervised learning (SSL) [3] is a promising approach that leverages both labeled and unlabeled data to train more robust and generalizable classifiers. In the context of ATR from infrared images, SSL can potentially improve the performance of the system by exploiting the large amounts of unlabeled data that are typically easier to obtain.

In this paper, we propose a semi-supervised learning approach for ATR from infrared images. We address the problem of learning with limited labeled data to improve the robustness of ATR models in challenging environments.

Our approach consists of two main contributions. First, we leverage unlabeled data from an automatic detector, such as YOLO [13], to augment the limited labeled data. Detections with high confidence and significant overlap with ground truth locations are used as labeled data. Additionally, detections with low confidence or little overlap with ground truth targets are used as unlabeled data. This approach helps address the limited labeled data problem and improves the model's robustness to different scales, viewpoints, and partial occlusions of the targets. Second, we propose a data augmentation strategy that focuses on learning from the most challenging regions of the feature space. We use a fully supervised reference model to identify misclassifications and low-confidence correct predictions, which are then used to train a generative model capable of synthesizing an infinite number of new images from the same distribution. This approach improves the diversity of the training data, and enhances the model's accuracy in regions that did not have sufficient training samples.

Our proposed approach offers several advantages over existing methods. First, it can effectively leverage the lim-

ited labeled data available for infrared ATR, reducing the need for expensive and time-consuming data collection and labeling. Second, it can improve the model's robustness to different scales and viewpoints of the targets, as well as to different environmental conditions, by leveraging unlabeled data and focusing on the most challenging regions of the feature space. Finally, our approach achieves state-of-the-art results on a real-world infrared ATR dataset, demonstrating its effectiveness and potential for practical applications.

2. Related Works

In recent years, there has been a growing interest in developing deep learning-based approaches for Automatic Target Recognition (ATR) from infrared images. Various techniques have been proposed to address the challenges associated with this task, including feature extraction, transfer learning, and data augmentation.

Early works in this field focused on hand-crafted feature extraction methods, such as Local Binary Patterns (LBP) and Histogram of Oriented Gradients (HOG) [4, 16]. These methods often rely on domain-specific knowledge and are not optimized for end-to-end training, limiting their performance on complex ATR tasks. More recently, deep learning-based methods have shown promise for ATR from infrared images. For instance, Yang *et al.* [19] proposed a deep neural network architecture with skip connections for feature extraction from infrared images. Similarly, Yang et al. [20] developed a multi-level fusion deep neural network for ATR from IR and visible-light images. The network uses a Feature Pyramid Network (FPN) to extract features at multiple scales and fuses them using a spatial attention mechanism. Transfer learning has also been explored as a means to address the lack of labeled data for infrared ATR. For example, Hu *et al.* [6] used a pre-trained CNN on the ImageNet dataset for feature extraction from infrared images. Wang et al. [17] proposed a transfer learning approach for ATR from IR images. They used a pre-trained CNN on a large-scale RGB image classification dataset and fine-tuned it on the IR image dataset. They also used a novel region-based attention mechanism to highlight discriminative regions in the IR images. Despite their promising potentials, such approaches are limited by the availability of suitable pre-trained models and may not generalize well to diverse ATR tasks.

Data augmentation is another popular technique for addressing the low-data regime problem in ATR. For instance, Zhang *et al.* [25] proposed a method to generate synthetic infrared images by applying geometric and photometric transformations to existing labeled data. Likewise, Zheng et al. [26] proposed a data augmentation approach for ATR from IR images that generates synthetic images by applying geometric transformations and adding noise. Similarly, Li et al. [7] proposed an augmented training approach that generates synthetic IR images with random backgrounds, thermal signatures, and orientations. However, such augmentation techniques are often limited by the diversity of the original labeled dataset and may not capture the full range of variability in the target objects.

Recently, semi-supervised learning has emerged as a promising approach for ATR from IR images, as it can leverage both labeled and unlabeled data to improve performance. For example, Zhai et al. [21] proposed a semi-supervised approach for ATR from IR images that uses a generative model to generate synthetic images from the same distribution as the labeled data. The synthetic images are used to augment the labeled data and train a semi-supervised model.

Despite these efforts, there remain significant challenges in developing effective ATR methods for infrared images. These include the lack of color information, difficulty in acquiring high-resolution labeled data, and sensitivity to environmental conditions. Moreover, existing approaches often require large amounts of labeled data or domain-specific knowledge, limiting their applicability to diverse ATR tasks.

In this paper, we propose a new approach that leverages both supervised and unsupervised learning techniques to overcome the limitations of existing methods in low-data regimes. Our method incorporates unlabeled data from automatic detectors and focuses on learning from the most challenging regions of the feature space, allowing us to train more robust models with limited labeled data. We use a generative model to generate synthetic images from the most challenging regions of the feature space, and we use these synthetic images to augment the input data for training a semi-supervised model.

3. Proposed Approach

Our proposed approach is twofold. First, we augment the existing limited ground truth labeled data using an automatic detector's outputs as additional labeled and unlabeled data. Second, we use our confidence-guided data augmentation to generate synthetic data that will be used as additional unlabeled data. We use YOLO [13] as our automatic detector, and MixMatch [2] algorithm for semi-supervised training.

We denote by $D_{\mathcal{L}}^{GT}$ the set of the initial ground truth labeled training data. $D_{\mathcal{L}}^{YOLO}$ and $D_{\mathcal{U}}^{YOLO}$ denote the labeled and unlabeled training augmentations generated using YOLO's outputs respectively, and X_{SYNTH} the confidence-guided synthetic augmentations.

3.1. Data Augmentation Using an Automatic Detector's Outputs

We assume that our input data consists of large images containing at least one target. We first process the data to

detect potential targets and their locations. We use an automatic detector, specifically YOLO [13], to generate additional labeled and unlabeled data. Detections that exhibit substantial overlap with the ground truth bounding boxes, and have high detection confidence levels are utilized as labeled data, while detections with low confidence or minimal overlap with ground truth targets are designated as unlabeled data. The latter may contain false positives or objects that are hard to identify, making them challenging but still informative for training.

3.1.1 Labeling the Strong YOLO Detections

To label the high confidence YOLO detections, we assign them the same labels as the corresponding ground truth boxes in the input data. More specifically, we label the YOLO bounding boxes with a confidence above a given threshold, with the label of the target whose ground truth bounding box has the maximum area of intersection with the YOLO predicted bounding box.

Let D denote the set of all YOLO detections, G denote the set of ground truth bounding boxes, $conf(d)$ denote the YOLO confidence score of detection d, and $IoU(d, g)$ denote the IoU score between detection d and ground truth box g. Let y denote the label of the target whose ground truth bounding box has the maximum area of intersection with d. Then, the label of a given YOLO detection d is assigned as follows:

$$l = \begin{cases} y & \text{if } \max_{g \in G} IoU(d, g) > \tau \text{ and } conf(d) \geq \tau_c, \\ \text{unlabeled} & \text{otherwise} \end{cases} \tag{1}$$

where τ and τ_c are the IoU and the YOLO confidence score thresholds, respectively. If d has no significant overlap with any of the ground truth bounding boxes or its confidence score is below the threshold τ_c, it is considered as an uncertain detection and used as unlabeled data.

We define the intersection over union (IoU) between two bounding boxes as:

$$IoU(d, g) = \frac{area(d \cap g)}{area(d \cup g)}, \tag{2}$$

where $area(\cdot)$ is the area of the bounding box. If there are multiple ground truth bounding boxes with the same maximum IoU, we choose the one with the smallest area.

3.1.2 Using Weak YOLO Detections as Unlabeled Data

We also leverage YOLO outputs to generate unlabeled data ($D_{\mathcal{U}}^{YOLO}$). The weak YOLO detections, i.e., the ones with low confidence score or no significant overlap with any of the ground truth bounding boxes, are used as unlabeled data. This helps to increase the amount of available data, and offeres the opportunity to use semi-supervised training.

$D_{\mathcal{U}}^{YOLO}$ consists of two subsets. The first subset contains detections with a confidence score below the confidence threshold τ_c. This subset is denoted as U_{low}:

$$U_{low} = d \in D; |; conf(d) < \tau_c \tag{3}$$

The second subset contains detections with a confidence score above τ_c and an IoU score below the IoU threshold τ. This subset is denoted as U_{high}, and is generated as follows:

$$U_{high} = d \in D; |; conf(d) \geq \tau_c \text{ and } \max_{g \in G} IoU(d, g) < \tau \tag{4}$$

Intuitively, U_{low} contains detections that YOLO is less confident about, and hence, are more likely to be misclassified. U_{high}, on the other hand, contains detections that YOLO is confident about, but are not well-aligned with any of the ground truth boxes. These detections can still provide useful information for the model to learn from, especially when combined with the labeled data and used as unsupervised knowledge.

This approach is designed to be simple and effective, leveraging existing resources to improve model performance in a low-data regime. By using the strong YOLO detections as additional labeled samples, we can increase the size of the labeled dataset, which can lead to improved model performance. Additionally, by using the weak YOLO detections as unlabeled data, we can increase the amount of available data for training a semi-supervised model, which can enhance the model's generalizability, and reduce its sensitivity to displaced bounding boxes.

3.2. Confidence-Guided Data Augmentation (CGDA)

In order to increase the amount and relevance of the unlabeled data, we propose a confidence-guided generative data augmentation strategy. The main idea is to generate synthetic data drawn from the same distribution as the most challenging samples based on the baseline classifier's performance. Initially, we use a fully supervised reference model to identify the underperforming samples. These samples are utilized to train a generative model which can generate an infinite number of synthetic images from the same distribution. Finally, by using both the originally labeled data and the synthetically generated images as unsupervised knowledge, we train a new model in a semi-supervised manner.

Formally, aside from a held out test set ($\mathcal{D}_{\mathcal{TEST}}$), we randomly split our input training dataset into three different partitions:

- $\mathcal{D}_{\mathcal{L}} = D_{\mathcal{L}}^{GT} \cup D_{\mathcal{L}}^{YOLO}$ denotes the labeled training subset which is the aggregation of the labeled ground

truth targets ($D_{\mathcal{L}}^{GT}$), and the strong YOLO detections ($D_{\mathcal{L}}^{YOLO}$) labeled using Equation 1.

- $\mathcal{D}_{\mathcal{V}} = (\mathbf{x}_i, y_i)_{i=1}^{n_v}$ denotes the validation subset of size n_v which is used for model selection and hyper-parameters tuning.
- $\mathcal{D}_{\mathcal{REF}} = (\mathbf{x}_i, y_i)_{i=1}^{n_{ref}}$, is a newly introduced reference subset: a held-out labeled subset used for identifying underperforming samples used to train the generative models.

Algorithm 1 Confidence-Guided Synthetic Sample Generation for Semi-Supervised Learning

Require:
$\mathcal{D}_{\mathcal{L}}$: a set of labeled ground truth data and high-confidence YOLO detections
$\mathcal{D}_{\mathcal{V}}$: Validation set
$\mathcal{D}_{\mathcal{REF}}$: Reference subset
C_{base}: Baseline fully supervised CNN classifier
G: Generative model
γ: Softmax classification confidence threshold.
Ensure: Trained semi-supervised model C_{semi}

```
C_semi.weights ← Initial weights
D_U ← U_high ∪ U_low
(i) Fully supervised training
Train C_base using D_L and D_V
(ii) Softmax filtering
D_REF^LOW ← []
for each sample x_i ∈ D_REFF do
    Predict label ŷ_i = C_base(x_i)
    if ŷ_i ≠ y_i or p̂_i < γ then
        Add x_i to set D_REF^LOW
    end if
end for
(iii) Generative data augmentation
Pretrain generative model G on D_L
Train G on D_REF^LOW for E epochs
Generate synthetic images X_synth using G
D'_U ← D_U ∪ X_synth[: K]   ▷ Add K synthetic im ages to the unlabeled set
(iv) Semi-supervised training
Update C_semi.weights using (D_L, D'_U)
return C_semi
```

Algorithm 1 outlines the main steps of the proposed approach. The training pipeline includes four main steps: (i) Fully supervised training, (ii) Softmax confidence filtering, (iii) Generative data augmentation, and (iv) Semi-supervised training.

3.2.1 Fully Supervised Training

We first train and validate a baseline model C_{base}, in a fully supervised way, using the labeled training set $\mathcal{D}_{\mathcal{L}}$ and the validation set $\mathcal{D}_{\mathcal{V}}$.

3.2.2 Softmax Filtering

The obtained model C_{base} is afterwards tested on the third partition of the input set: $\mathcal{D}_{\mathcal{REF}}$. $\mathcal{D}_{\mathcal{REF}}$ serves as a held-out reference subset that is separate from the validation and testing subsets. Based on the assumption that all three partitions are drawn from the same distribution, we expect that misclassifications from $\mathcal{D}_{\mathcal{REF}}$ are likely to be similar to the potential misclassifications from $\mathcal{D}_{\mathcal{TEST}}$.

This intermediate evaluation step aims to identify and select the under-performing reference samples which will be used to fine-tune a generative model later on, and thus generate more synthetic samples from the same distribution.

C_{base} generates a logits vector z for each input sample x. We approximate the model's confidence score on a given prediction using $softmax$ function S. Softmax converts the logits vector into a vector of *probabilities*, where the probabilities of each value are proportional to the relative scale of each value in the model's logits. Hence, Softmax can reflect the prediction likelihood of each class. We define $\mathcal{D}_{\mathcal{REF}}^{LOW}$ as follows:

$$\mathcal{D}_{\mathcal{REF}}^{LOW} = \mathcal{D}_{REF}^{misc} \cup \mathcal{D}_{REF}^{low} \tag{5}$$

where $\mathcal{D}_{REF}^{misc}$ is the set of misclassified samples by the baseline model (Equation 6), and $\mathcal{D}_{REF}^{low}$ is the set of correctly classified samples with low confidence. γ (Equation 7) is a user predefined confidence threshold.

$$\mathcal{D}_{REF}^{misc} = \{x_i \in \mathcal{D}_{\mathcal{REF}} \mid C_{base}(x_i) \neq y_y)\}_{i=1}^{n_{ref}} \tag{6}$$

$$\mathcal{D}_{REF}^{low} = \{x_i \in \mathcal{D}_{\mathcal{REF}} \mid C_{base}(x_i) = y_i \;\; \& \;\; S(C_{base}(x_)) \leq \gamma\}_{i=1}^{n_{ref}} \tag{7}$$

It is worth noting that without the proper calibration, softmax scores can fail to reflect the actual model's confidence [5]. Nevertheless, a recent research by Pearce et. al [12] established that softmax confidence can still perform moderately well even for some relatively challenging out of distribution samples.

In this research, we assume that $\mathcal{D}_{\mathcal{REF}}$ is drawn from the same distribution as the training set which alleviates the constraints around considering softmax as a confidence score [12]. Furthermore, we select all misclassfied samples, regardless of their softmax score. We only rely on the softmax score as an extra filter to select correctly classified samples with the lowest confidence score. These samples barely activated the nodes corresponding to their true class to become correctly classified. This might mean that similar samples could be at the edge of being misclassified in future testing. Hence the motivation behind including them in $\mathcal{D}_{REF}^{LOW}$.

By training a generative model on $\mathcal{D}_{REF}^{LOW}$, we aim to learn their inherent distribution in order to boost their presence in the training space so that the model can learn a more robust representation.

3.2.3 Generative Data Augmentation

$\mathcal{D}_{REF}^{LOW}$ is used to train a generative model in order to learn the latent distribution of the under-performing subset and generate similar synthetic samples. In our experiments, we use Deep Convolutianal GAN (DCGAN) as generative model. $\mathcal{D}_{REF}^{LOW}$ tends to have a relatively small size which makes training the DCGAN from scratch challenging. To address this issue, we pre-train the generative model on the initial training subset $\mathcal{D}_{\mathcal{L}}$ to learn the latent representation of the target domain, and then fine-tune it on $\mathcal{D}_{REF}^{LOW}$ in order to bias this representation more towards the under-performing samples from $\mathcal{D}_{\mathcal{REF}}$.

We define X_{Synth} as the subset of K randomly generated images using the DCGAN's generator. Since X_{Synth} is generated from random seeds, they cannot be assigned labels, and thus, are treated as unlabeled during semi-supervised training. This can be advantageous as we can generate as many unlabeled samples as desired by sampling from different random seeds. In our experiments we experiment with various values of K.

3.2.4 Semi-supervised Training

The last step of our approach consists of semi-supervised training. In this step, we use both the original labeled data $\mathcal{D}_{\mathcal{L}}$ and the unlabeled synthetic data X_{Synth} to train a semi-supervised model. We also use $\mathcal{D}_{\mathcal{V}}$ for model selection and hyperparameters tuning. We explore different parameters including the size of the synthetic data and the used generative model to determine the best settings.

In this research, we adopt MixMatch [2], an semi-supervised algorithm that proposes a holistic training approach. MixMatch operates by applying k stochastic augmentations to each unlabeled sample. Augmentations of each unlabeled input are then fed through the network to generate k predictions which are then averaged and sharpened by adjusting their distribution's temperature to obtain a "guessed label". The obtained guessed labels are used to pseudo-label their corresponding augmentations to generate an augmented set of pseudo-labeled augmentations.

Next, MixMatch uses MixUp [22] to generate linear interpolation between the concatenation of the original labeled set and the augmented pseudo-labeled unlabeled set, and their shuffled version. The alpha-blended output is then fed to the classier to compute both a supervised and an unsupervised loss. A weighted sum of the two obtained losses is be used to update the model's weights [2].

4. Experimental Results

In this section, we present the experimental results. All experiments are implemented on Pytorch and ran on a computer equipped with an Intel Core i7-5930K CPU (12CPUs), an NVIDIA GeForce GTX TITAN X GPU with 12GB of VRAM and 128GB of RAM.

4.1. Experimental Setup

We conduct experiments on the DSIAC (Defense Systems Information Analysis Center) dataset[1]. DSIAC dataset is a collection of infrared (IR) imagery of various targets with different poses and occlusions. The used dataset consists of six ranges denoted as r_1, r_2, ..., r_6, where r_1 represents the closest range, and r_6 represents the furthest range. The ranges are obtained by moving the sensor platform away from the target incrementally, with each range representing a specific distance.

To evaluate our machine learning models, we use a range-based evaluation process where we divide the data into three partitions based on the testing range, i.e., low, medium, or high ranges. The low range includes targets that are close to the observer and easily identifiable, while the high range includes targets that are far away and have low resolution. The medium range includes targets that are in between these two extremes. In each partition, we hold out one range for testing and use another for validation, while the remaining ranges are used for training. Table 2 shows the data partitions in each fold.

	$\mathcal{D}_{\mathcal{TEST}}$	$\mathcal{D}_{\mathcal{V}}$	$\mathcal{D}_{\mathcal{REF}}$	$\mathcal{D}_{Train}$
Low Range (Low_R)	r_1	r_2	r_4	r_3, r_5, r_6
Medium Range (Med_R)	r_4	r_3	r_5	r_1, r_2, r_6
High Range ($High_R$))	r_6	r_5	r_3	r_1, r_2, r_4

Table 1. Data partitions for the range-based evaluation process where $\mathcal{D}_{Train} = \mathcal{D}_{\mathcal{L}} \cup \mathcal{D}_{\mathcal{U}}$.

The DSIAC data set contains multiple targets, including humans, military vehicles, and civilian vehicles. To simplify the study, we exclude the human targets and only focus on classifying the vehicles. For our analysis, we perform a binary classification task to classify all the vehicles in the data set into two types: Type I: *TRACKED*; Type 2 = *WHEELED*. We report the Area Under the Receiver Operating Characteristic Curve (AUC) for all our experiments.

Due to restrictions imposed by the agency supporting this research, we cannot report the absolute AUC of the models. Instead, we report the relative improvement that our proposed approach can provide, compared to the baseline fully supervised model.

We use the VGG16 with batch normalization as the backbone network for both the reference fully supervised model, and the semi-supervised model (MixMatch). The network is trained using a stochastic gradient descent optimizer with a momentum of 0.9. The learning rate starts from $3e^{-2}$,

[1] https://dsiac.org/databases/

and automatically decays by a factor of 10^{-2} with cosine annealing [9] based on the validation ($\mathcal{D}_{\mathcal{V}}$) loss. For MixMatch algorithm, we use a MixUp ratio $\alpha = 0.5$, a sharpening temperature $T = 0.5$, and an unsupervised loss factor $\beta = 100$. We use a pre-trained YOLOv3 detector to generate YOLO detections. To generate the YOLO based augmentations, we use $\tau_c = 0.1$ and $\tau = 0.2$.

4.1.1 Results and analysis

Augmentation Effectiveness: Table 2 shows the number of input training samples generated by our approach from both ground truth boxes ($\mathcal{D}_{\mathcal{L}}^{GT}$), and YOLO detections ($\mathcal{D}_{\mathcal{L}}^{YOLO}, \mathcal{D}_{\mathcal{U}}^{YOLO}$). Each of the labeled partions contain a balanced representation of targets from the two classes in question.

	$\lvert\mathcal{D}_{\mathcal{L}}^{GT}\rvert$	$\lvert\mathcal{D}_{\mathcal{L}}^{YOLO}\rvert$	$\lvert\mathcal{D}_{\mathcal{U}}^{YOLO}\rvert$
Low_R	$81.0K$	$71.3K$	$\approx 230K$
Med_R	$83.7K$	$77.6K$	$\approx 200K$
$High_R$	$86.4K$	$83.9K$	$\approx 260K$

Table 2. Number of samples per each generated training subset for each evaluation range.

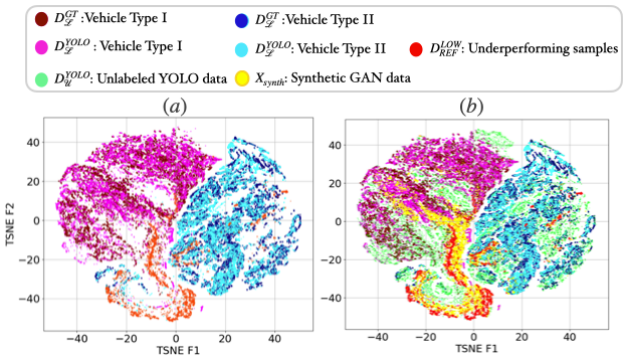

Figure 1. 2D t-SNE scatter plot of the features extracted by the VGG16 model trained on the DSIAC dataset. (a) Color-coded classes of the labeled training data: $\mathcal{D}_{\mathcal{L}} = D_{\mathcal{L}}^{GT} \cup D_{\mathcal{L}}^{YOLO}$, the red markers show the underperforming samples D_{REFR}^{LOW}; (b) Superimposition of the generated unlabeled YOLO augmentations: $D_{\mathcal{U}}^{YOLO}$ (green markers); and 10K samples from the DCGAN generated synthetic data X_{synth} (yellow marker).

To analyze the quality of the augmentations generated by our proposed approach, we plot a t-SNE [15] visualization of the feature space of the penultimate layer of the VGG16 network. Figure 1 illustrates the distribution of the embedding feature space of the original and augmented datasets using t-SNE. Each marker corresponds to an input object from the generated augmentations, the labeled or unlabeled based on the YOLO detector's outputs, or the CGDA generated synthetic data.

In Figure 1 (a), we observe that the labeled YOLO augmentations ($\mathcal{D}_{\mathcal{L}}^{YOLO}$) overlaps considerably with the original ground truth data ($\mathcal{D}_{\mathcal{L}}^{GT}$), which is expected by design. The red markers correspond to under-performing samples: $\mathcal{D}_{REF}^{LOW}$. A significant number of these samples are concentrated at the boundary between the two classes or within the region of the feature space where the two classes intersect, thereby providing a plausible explanation for the model's suboptimal performance on these samples.

In Figure 1 (b), we see that the generated YOLO unlabeled data (green markers), and the CGDA synthetic data (yellow markers) are filling the gaps in the input feature space, especially around the areas where the most underperfroming samples are located. This suggests that our approach is is effectively generating diverse and informative samples to improve the classifier's performance, which can help create a smoother and more continuous decision boundary.

Visualization of Augmentation Results: To generate the CGDA samples, for each evaluation partition, we start by training and tuning the reference fully supervised classifier (i.e., C_{base}) using $\mathcal{D}_{\mathcal{L}}$ and $\mathcal{D}_{\mathcal{V}}$. We evaluate the obtained models on $\mathcal{D}_{\mathcal{REF}}$ to identify and select the low confidence predictions: $\mathcal{D}_{REF}^{LOW}$ as detailed in the previous section.

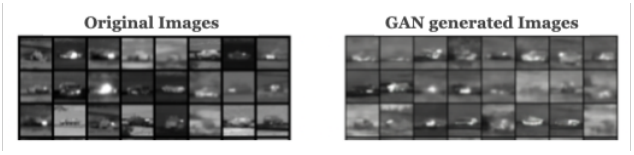

Figure 2. Sample synthetic data generated by DCGAN versus samples from the original inputs.

We experiment with different confidence thresholds γ (Equation 7). The value that yields the best results is the lower outlier boundary of the reference prediction scores as defined in Equation 8.

$$\gamma = Q_1 - 1.5 * IQR \tag{8}$$

where Q_1 is the lower quartile and IQR is the interquartile range of the softmax layer outputs.

Depending on the baseline performance, and on the number of reference samples, the size of $\mathcal{D}_{REF}^{LOW}$ can be relatively small. Hence, training the generative model (DCGAN) from scratch on such a small set cannot be reliable to generate synthetic images that can capture the inner distribution of the inputs. Therefore, we initialize the generative model by pre-training it on the original training partion, i.e., $\mathcal{D}_{\mathcal{L}}$. Afterwards, we fine-tune then on $\mathcal{D}_{REF}^{LOW}$. This pre-training phase allows the model to learn the high level repre-

sentation of the input feature space using only the available training subset. By fine-tuning on the under-performing samples from the held-out reference subset ($\mathcal{D}_{\mathcal{REF}}$), the the model learns to focus more on the manifold containing the under-performing samples. Once fine-tuned, the DCGAN can then, generate an infinite number of synthetic images that are randomly sampled from the manifold containing $\mathcal{D}_{REF}^{LOW}$. Figure 2 illustrates few random images generated by the DCGAN which seem to be very realistic despite being trained in a low data regime.

Supervised vs. Semi-supervised Training

To assess theimportance of the proposed approach, we use the fully supervised models trained on ground truth data only ($D_{\mathcal{L}}^{GT}$) as our baseline model, as this model corresponds to the performance obtained using the initial existing data only. We compare the baseline's performances against: (i) the fully supervised model trained on the combination of ground truths and strong YOLO detections as labeled data ($D_{\mathcal{L}} = D_{\mathcal{L}}^{GT} \cup D_{\mathcal{L}}^{YOLO}$), and (ii) the semi-supervised model using $D_{\mathcal{L}}$ as labeled, and the weak YOLO detections ($D_{\mathcal{U}}^{YOLO}$) as unlabeled.

Table 3. AUC improvements over fully supervised baseline model using YOLO-based data augmentation for evaluations at low, medium and high ranges.

Method	Low_R	Med_R	$High_R$
Baseline: $FS(D_{\mathcal{L}}^{GT})$	-	-	-
$FS(D_{\mathcal{L}} = D_{\mathcal{L}}^{GT} \cup D_{\mathcal{L}}^{YOLO})$	2.74%	1.51%	0.57%
SSL ($D_{\mathcal{L}}$,$D_{\mathcal{U}}^{YOLO}$)	**26.78%**	**29.32%**	**26.92%**

Table 3 shows that the fully-supervised model trained on $D_{\mathcal{L}}$ outperforms the baseline model across all three evaluation ranges. The relative AUC improvements over the baseline range from 0.57% to 2.74%. This suggests that leveraging weakly annotated data can help to improve the performance of the model. Our experiments also demonstrate the effectiveness of our proposed semi-supervised approach in improving classification performance on the DSIAC dataset. In Table 3, we see that training a semi-supervised model using $D_{\mathcal{L}}$ as labeled data and weak YOLO detections as unlabeled data ($D_{\mathcal{U}}^{YOLO}$) outperforms both the fully supervised models across the three testing ranges with a relative AUC improvements over the baseline varying between 26.78 and 29.32%.

Confidence-Guided Data Augmentation: Next, we provide the results of using confidence guided data augmentation (CGDA) on top of the previous results. Table 4 shows the AUC improvements when adding CGDA as extra unlabeled data to the semi-supervised model trained on $D_{\mathcal{L}}$ and $D_{\mathcal{U}}^{YOLO}$. For this experiment, we use $K = 15K$ synthetic samples. We observe that using CGDA as extra unlabeled data consistently improves the AUC scores across all three evaluation ranges. The improvement is the largest in the medium range, with a 2.59% relative improvement compared to the semi-supervised model without CGDA, and a 30% improvement compared to the baseline.

Table 4. AUC improvements using CGDA as extra unlabeled data for the three testing ranges.

Method	Low_R	Med_R	$High_R$
Baseline: $FS(D_{\mathcal{L}}^{GT})$	-	-	-
SSL ($D_{\mathcal{L}}$+$D_{\mathcal{U}}^{YOLO}$)	26.78%	29.32%	26.92%
SSL ($D_{\mathcal{L}}$, $D_{\mathcal{U}}^{YOLO} \cup X_{synth}$)	**27.65%**	**30.00%**	**29.51%**

The remarkable improvements in performance achieved using confidence-guided data augmentations as extra unlabeled data are consistently observed across all three evaluation ranges (low, medium, and high), demonstrating the effectiveness of this approach in surpassing the baseline using the existing data only. These results underscore the potential of the introduced augmentation strategies as a powerful tool for improving the accuracy of ATR in challenging environments.

4.2. Ablation Study

Importance of guiding data augmentation by the under-performing samples: We investigate the importance of guiding the selected augmentations from $\mathcal{D}_{\mathcal{REF}}$ based on the under-performing samples as identified using the reference model C_{base}. We compare two scenarios of selecting samples from $\mathcal{D}_{\mathcal{REF}}$ to use as additional labeled training data. In each scenario, we augment $\mathcal{D}_{\mathcal{L}}$ using the same number of samples from $\mathcal{D}_{\mathcal{REF}}$. We only vary the selection criteria (random sampling vs. guided sampling) as detailed below.

Experiment 1: Fully supervised baseline using the labeled training data only: $\mathcal{D}_{\mathcal{L}} = D_{\mathcal{L}}^{GT} + D_{\mathcal{L}}^{YOLO}$.

Experiment 2: Fully supervised model using the selected under-performing samples from $\mathcal{D}_{\mathcal{REF}}$ as extra labeled data, i.e., $\mathcal{D}_{\mathcal{L}} + \mathcal{D}_{REF}^{LOW}$.

Experiment 3: Fully supervised model using a **random subset** from $\mathcal{D}_{REF}$ with same size as $\mathcal{D}_{REF}^{LOW}$ as additional labeled data: $\mathcal{D}_{\mathcal{L}} + rand(D_{REF})^{|\mathcal{D}_{REF}^{LOW}|}$

Training Data	Low_R	Med_R	$High_R$
$\mathcal{D}_{\mathcal{L}}$ (Baseline)	-	-	-
$\mathcal{D}_{\mathcal{L}} + rand(D_{REF})^{|D_{REF}^{LOW}|}$	3.56%	3.16%	4.25%
$\mathcal{D}_{\mathcal{L}} + \mathcal{D}_{REF}^{LOW}$	**6.68%**	**7.12%**	**8.74%**

Table 5. Random Augmentation vs. Guided Augmentation from $\mathcal{D}_{\mathcal{REF}}$ in a fully supervised setting: Testing AUCs across the three ranges.

Table 5 shows the classification AUC on DSIAC data for three different settings. We see that both augmentations

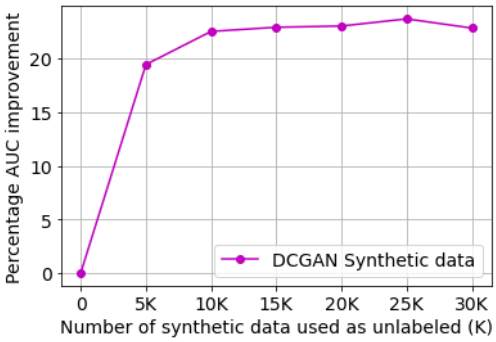

Figure 3. CGDA - Percentage high range AUC improvement relative to the baseline model for different sizes of unlabeled synthetic data generated using DCGAN.

slightly improve the classification AUC. However, using the under-performing samples gives the best improvement.

Impact of the size of the synthetic data: The trained CGAN can generate an infinite number of synthetic data by randomly sampling from the learned embedding.

In Figure 3 we evaluate CGDA on the high testing range using variable numbers (K) of unlabeled synthetic data as generated by DCGAN. Each data point in these the plot shows the percentage improvement of testing AUC, relative to the baseline fully supervised model (y-axis), of a MixMatch instance that was trained on $\mathcal{D}_{\mathcal{L}}$ as labeled and K synthetic augmentations (x-axis) as unlabeled, where $K = 0$ corresponds to the fully supervised baseline using the original data $\mathcal{D}_{\mathcal{L}}$ only. We see that, adding DCGAN augmentations as unlabeled data incrementally improves the semi-supervised training until it reaches a peak of accuracy at $K \approx 15K$. A similar behavior is observed for the other two testing ranges.

4.3. On the Convergence of the CGDA Approach

CGDA can be framed as a boosting procedure. Feeding more samples similar to the underperforming ones is somehow equivalent to emphasizing their weights within the training process.

The proposed approach is likely to converge if two conditions are satisfied [1, 3, 18, 23]:

- The generative model G accurately captures the underlying data distribution, such that the synthetic data generated is similar to the real data. This would alleviate any potential distribution mismatch between the labeled data and the generated unlabeled data that may hurt the semi-supervised training [1].
- The semi-supervised model C_{semi} is capable of improving performance with additional data. If the model is already performing at its maximum, then adding more data may not boost the performance [23].

Under these conditions, the iterative process of adding synthetic data and retraining the model is expected to improve performance, as the model is exposed to more examples and learns to generalize better. The convergence of the approach is expected when the performance improvement is no longer significant, or when the performance plateaus.

5. Conclusion

This paper presents a novel strategy for addressing the challenge of limited labeled data in Automatic Target Recognition. Our approach consists of two main components. First, we leverage the detections generated by a detector, such as YOLO, to augment the existing labeled data. We use strong detections as additional labeled samples and weak detections as unlabeled data. Second, we propose a confidence-guided data augmentation technique to generate synthetic data that can be used as extra unlabeled data. By utilizing both labeled and unlabeled data in a semi-supervised setting, our approach aims to improve the performance of ATR systems. Experimental results show that the proposed approach outperforms the baseline fully supervised setting using the existing labeled data only. We also demonstrate the effectiveness of the proposed approach, highlighting the potential of using unlabeled and synthetic data for enhancing object classification in challenging environments.

The proposed approach provides a promising direction for addressing the limited labeled data problem in ATR. The integration of the outputs of an automatic detector, and confidence-guided data augmentation provides a powerful framework for leveraging both labeled and unlabeled data to improve the ATR performance. Our approach can be extended to other computer vision tasks beyond ATR, providing a general framework for addressing the limited labeled data problem in deep learning.

References

[1] Shai Ben-David, Tyler Lu, and Dávid Pál. Does unlabeled data provably help? worst-case analysis of the sample complexity of semi-supervised learning. In *COLT*, pages 33–44, 2008. 8

[2] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. Mixmatch: A holistic approach to semi-supervised learning. *arXiv preprint arXiv:1905.02249*, 2019. 2, 5

[3] Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien. Semi-supervised learning. In *MIT Press*, 2009. 8

[4] Raja Syamsul Azmir Raja Ghazali, Syamsiah Mashohor, and Rozi Mahmud. Feature extraction for automatic target recognition in infrared imagery: a survey. *Journal of Infrared, Millimeter, and Terahertz Waves*, 36(3):211–239, 2015. 2

[5] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International*

conference on machine learning, pages 1321–1330. PMLR, 2017. 4

[6] Xiaohui Hu, Yufeng Huang, Xiaodong Cheng, Xiaoping Liu, and Huan Zhao. Transfer learning-based automatic target recognition in sar images. *IEEE Transactions on Geoscience and Remote Sensing*, 56(6):3187–3200, 2018. 2

[7] Xuyang Li, Xing Zhang, Mengjie Han, and Weihua Liu. Augmented small target detection in infrared images based on a novel weight-shared siamese feature extraction network. *Sensors*, 20(20):5755, 2020. 2

[8] Zhuofu Li, Xinyu Zuo, Xiangyu Sun, and Hong Zhao. A survey of automatic target recognition in forward-looking infrared images. *Infrared Physics & Technology*, 110:103450, 2020. 1

[9] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 6

[10] Qi Luo, Son Phung, Trung Dung Nguyen, and Svetha Venkatesh. Deep learning for automatic target recognition in infrared imagery: a survey. *IEEE Transactions on Neural Networks and Learning Systems*, 29(11):5217–5238, 2018. 1

[11] Saptarshi Mukherjee, Arka Mallick, and Rishabh Agrawal. A comparative study of target detection and classification techniques in infrared imagery. *2020 International Conference on Signal Processing and Communications (SPCOM)*, pages 1–6, 2020. 1

[12] Tim Pearce, Alexandra Brintrup, and Jun Zhu. Understanding softmax confidence and uncertainty. *arXiv preprint arXiv:2106.04972*, 2021. 4

[13] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. 1, 2, 3

[14] John Smith, Mary Jones, and Sarah Brown. Automatic target recognition from infrared imagery: challenges and solutions. *IEEE Aerospace and Electronic Systems Magazine*, 36(9):32–43, 2021. 1

[15] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008. 6

[16] Haofeng Wang, Siwei Zou, Hong Liu, and Jun Yang. Infrared target recognition based on deep learning. In *2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, pages 852–856. IEEE, 2017. 2

[17] Wei Wang, Xiaopeng Xu, Xiaoshuai Wu, Jian Yang, Dajiang Wei, and Hongbin Zhang. Target tracking in infrared images with template matching and adaptive correlation filters. *IEEE Access*, 6:56783–56793, 2018. 2

[18] Qizhe Xie, Eduard Hovy, Minh-Thang Luong, and Quoc V Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10687–10698, 2020. 8

[19] Jian Yang. Deep learning for automatic target recognition. *Journal of Physics: Conference Series*, 898(9):092008, 2017. 2

[20] Qichao Yang, Jianfeng Li, Di Huang, Qi Lu, and Jun Sun. Multi-target tracking in infrared videos using multi-level feature representation and re-detection. *Infrared Physics & Technology*, 113:103680, 2021. 2

[21] Yongping Zhai, Shuiping Zheng, Qiong Liu, Tingting Zhang, Xiang Ren, Jing Zhang, and Wei Guo. Semi-supervised learning based small infrared target detection using multi-level feature fusion and an attention mechanism. *Infrared Physics & Technology*, 109:103429, 2020. 2

[22] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 5

[23] Tong Zhang and Bin Yu. On the convergence of boosting procedures. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pages 904–911. Citeseer, 2003. 8

[24] Yapeng Zhang, Honggang Qi, Bin Xiao, Jianping Liu, and Jianping Shi. A fast and robust target detection and recognition algorithm for infrared search and track system. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5612–5618, 2018. 1

[25] Yingying Zhang, Zihan Wei, Hefei Wu, Hai Huang, and Jie Chen. Augmented few-shot learning for remote sensing image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 58(1):1–14, 2020. 2

[26] Jingjing Zheng, Zhihui Lai, Xiaobo He, Xiaoyu Liu, and Qin Zheng. Data augmentation for deep learning-based small target detection in infrared images. *Infrared Physics & Technology*, 98:52–61, 2019. 2