# Long-Tailed Recognition of SAR Aerial View Objects by Cascading and Paralleling Experts

Cheng-Yen Yang, Hung-Min Hsu, Jiarui Cai, Jenq-Neng Hwang
Department of Electrical and Computer Engineering, University of Washington
cycyang, hmhsu, jrcai, hwang @ uw.edu

## Abstract

*Aerial View Object Classification (AVOC) has started to adopt deep learning approaches with significant success in recent years, but limited to optical data. On the other hand, Synthetic Aperture Radar (SAR) has wild aerial view related applications in the remote sensing field. However, SAR has received far less attention due to the special characteristics of the SAR data, which is the long-tailed distribution of the aerial view objects that increases the difficulty of classification. In this paper, we present a two-branch framework, including the cascading expert branch and paralleling expert branch, to tackle the long-tailed distribution of the dataset. Our proposed multi-expert architecture achieves 24.675% and 26.029% in the development phase and testing phase, respectively, in the NTIRE 2021 Multi-modal Aerial View Object Classification Challenge Track 1. The proposed method is proved to possess the effectiveness (top-tier performance among 157 participants) and efficiency (*i.e*., a lightweight architecture) for the AVOC task.*

## 1. Introduction

In the past few years, deep learning has attracted tremendous attention due to its impressive feature representation capability in a large amount of computer vision and pattern recognition applications. Using Electro-Optical (EO) sensors for Aerial View Object Classification (AVOC) has been the most prevalent approach since the captured images are represented in RGB and gray-scale images. However, another sensor, called Synthetic Aperture Radar (SAR), can provide high-resolution radar frequency (RF) images. Comparing to EO sensors, SAR is able to capture significant information under various scenarios, (*i.e*. weather conditions, no visible light, *etc*.). In this case, SAR can be used to complement EO sensors by reproducing the RF images. Therefore, the motivation for this work is to investigate how SAR can be used to improve the classification performance for AVOC.

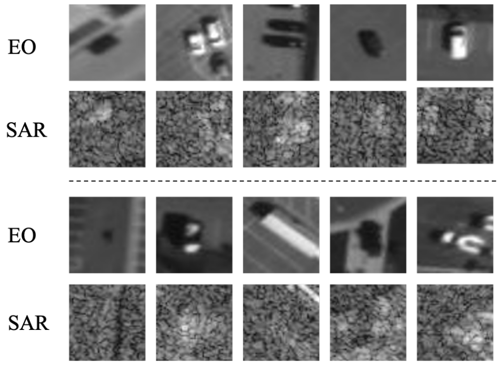

Figure 1: Samples of EO and SAR images for each class from the NTIRE2021 challenge training dataset. From left to right, top row represents sedan, suv, pickup truck, van and box truck while lower row represents motorcycle, flatbed truck, bus, pickup truck with trailer and flatbed truck with trailer.

The purpose of AVOC is to predict the class label of a low-resolution aerial image. Due to the success of convolutional neural networks (CNNs) for various pattern recognition tasks, most researchers use CNNs as the feature extractor, then exploit the deep discriminative features of SAR images for classification [27]. VGG [18] and VGG-S net [1] are used as the backbone for SAR image feature extraction. Moreover, a deep sparse tensor filtering network (DSTFN) [36] is proposed to generate more discriminative features by using the filters from a least squares support vector machine.

On the other hand, the automatic target recognition (ATR) method [17], which uses principal component analysis (PCA) features, elliptical Fourier descriptors (EFDs), and azimuthal sensitivity image (ASI) as the complementary features, is proposed to describe the SAR target for classification.

According to the previous works, VGG based networks are wildly used in SAR image classification and sparse tensor filtering is applied to generate more robust features. Therefore, there are two significant characteristics in AVOC: (1) too complicated architectures are not suitable for

SAR images since SAR images are low-resolution, where complex architecture is easily to be over-fitting, (2) the performance is easily affected by the sparse data, which are due to the imbalanced data distribution [23]. Thus, to solve these two issues, we aim to propose multiple shallow architecture models to deal with different data sizes of classes.

In this paper, we propose a novel multi-expert perception classifier including three types of experts: Routing Diverse Experts (RIDE) [33], ResNet-50 [15] based classifier and ResNet-50 based tail classifier. Since RIDE is already a multiple classifier framework, the intuition is to consider a framework with more trained models for different number of classes to enhance the robustness of the multi-expert framework.

Based on the Track 1 SAR imagery dataset of the NTIRE 2021 Challenge [21], we can achieve 24.675% and 26.029% in validation phase test data and final test data, respectively. The promising experimental results show that the proposed method can possess the effectiveness (top-tier performance among 157 participants) and efficiency (i.e., a lightweight architecture) for AVOC.

Overall, the main contributions of this paper can be summarized as follows,

- A novel multi-expert perception classifier is proposed for the AVOC task.
- Using trained models for different number of classes to improve the accuracy of AVOC task.
- A light-weighted architecture is used to ensure efficiency.
- The accuracy archives the top-tier among all the teams.

The rest of this paper is organized as follows. Related works are presented in Section 2, and the proposed method is introduced in Section 3. In Section 4, we evaluate the proposed method on the Track 1 SAR imagery dataset of the NTIRE 2021 Challenge [21]. Finally, we draw a conclusion in Section 5.

## 2. Related Works

**Synthetic Aperture Radar** Due to the success of deep learning in the computer vision and pattern recognition communities, the growing use of deep learning in Synthetic Aperture Radar (SAR) attracts much attention [41] such as terrain surface classification [26], object detection [6], parameter inversion [31], despeckling [32], specific applications in Interferometric SAR (InSAR) [2], and SAR-optical data fusion [16]. In terms of SAR target detection, traditional approaches mainly focus on template matching based on the handcrafted feature via traditional machine learning approaches, such as Support Vector Machines (SVMs) [38, 3]. In contrast, deep learning algorithms use end-to-end convolutional neural networks (CNNs) as the feature extractor to generate discriminative features that can seamlessly work with the subsequent classifier automatically [6]. In this work, we aim to use SAR for the AVOC task to compensate the EO sensors by the RF images. To the best of our knowledge, there is no multi-expert deep learning framework to solve the long-tailed distribution of SAR images in AVOC.

**Aerial View Object Classification** Generally, the methods for SAR image recognition adopt the conventional CNN models [15] for natural images. However, since the SAR images are low in resolution and noisier, the literatures propose multiple methods for effective feature extraction and training. Early works utilize pre-processed manually-designed features [30, 11, 10], which are not optimal for training the networks end-to-end. Cho *et al.* [7] use multiple extraction branches to generate smoother features. To address the data limitation issue, Shang *et al.* [28] propose a densely connected and depth-wise separable CNN (DSNet) structure, which can reuse the hierarchical feature maps and avoid extracting redundant features; Min *et al.* [24] use model distillation to reduce model parameters. Nowadays, the data imbalance problem becomes the new bottleneck for SAR recognition. Previous research [29] finds naive over-sampling or under-sampling methods are not sufficient to solve the problem alone. Further exploration is required in both network architecture and re-balancing techniques.

**Long-Tailed Recognition** The methods for long-tailed recognition (LTR) can be divided into three categories: data re-balancing and augmentation, two-stage training, and multi-expert architectures. Data re-balancing mainly consists of (1) re-sampling, including down-sampling of the majority classes [12, 22] or over-sampling of the minority classes [5, 14, 25]; (2) re-weighting, which assigns higher weights on tails [9] or hard samples [4] in the loss function. Besides, data augmentations by mix-up [37, 8] or sample synthesis [37, 20] can also compensate for the insufficiency of training data and improve the generalizability of the model. However, such methods usually increase the performance of the tails by sacrificing that of the heads, indicating the under-representation of the majority classes. Therefore, later literatures widely adopt a two-stage training scheme [19] that trains the model with instance-balanced sampling (imbalanced) as a pre-training stage, followed by re-adjusting the classifier with a balanced sampled set. Though the overall performance is improved by a large margin, the use of re-balancing still hurts the accuracy of heads, and the re-weighting techniques are sensitive to hyper-parameters selection. More recently, multi-expert architectures show a strong capacity to address the LTR by aggregations of models. BBN [39] is a bilateral-branch network containing a uniform-sampled branch and a reversed-sampled branch. With a cumulative learning strategy, BBN is an end-to-end framework combining the afore-

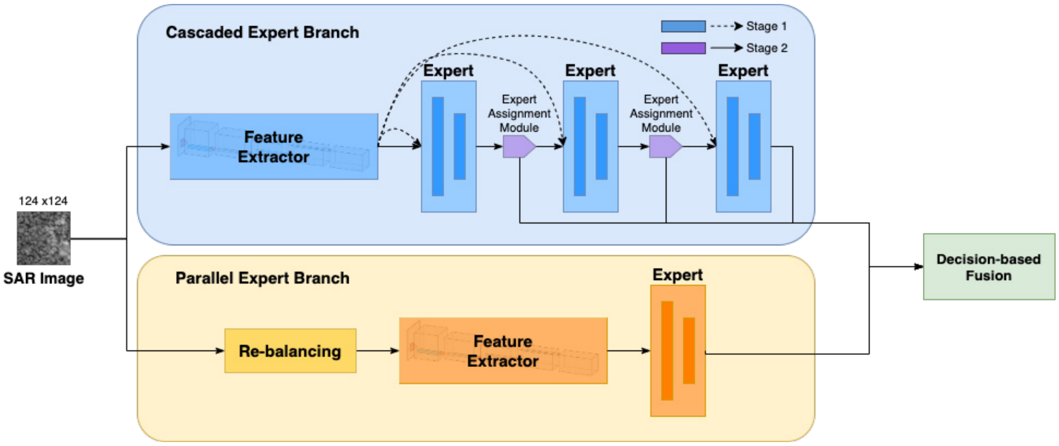

Figure 2: The overall framework of our proposed method. For each model in the cascaded expert branch, we first train the feature extractor and the experts in the first stage (marked as blue blocks and dashed arrows), followed by training the expert assignment modules in the second stage (marked as purple blocks and solid arrows). For each model in the parallel expert branch, we used a re-balancing method to train each individual expert. Finally, a weighted decision-based fusion is applied to get the final predictions.

mentioned two-stage scheme. LFME [35] assigns different data splits to the experts and trains each expert with knowledge distillation, thus has accuracy gain over all splits. The most recent state-of-the-art approach, RIDE [34], is a boosting method that uses KL-divergence loss to encourage the experts to be good at different parts of the dataset and learns to decide the number of experts used automatically by a designed label for routing.nsde

# 3. Method

## 3.1. Overall Framework

The long-tailed distribution of this NTIRE 2021 Multi-modal Aerial View Object Challenge Dataset is not typical as one of its head-classes even significantly out-numbered the other head-classes, which will be described later in Section 4.1. To tackle this atypical dataset, special attention is paid on designing the proposed deep learning strategy.

The overall framework of our proposed method can be separated into two components, the cascaded expert branch and parallel expert branch. For the cascaded expert branch, the experts are sequentially routed and each of the expert is trained using the entire dataset, resulting in making better predictions on the head-classes. Therefore, we introduce the parallel expert branch into the architecture as it uses re-balancing method in the training process to alleviate the influence of the imbalanced sample distributions. Finally, a decision-based voting fusion is performed using the outputs from the two branches.

## 3.2. Cascaded Expert Branch

For our cascaded expert branch, we adopt the current state-of-the-art method, the two-stage framework RountIng Diverse Experts (RIDE) [33], to to gain performance on tail classes without sacrificing head-classes accuracies in the LTR tasks.

In the first stage, we train the shared feature extractor and $n$ experts, which are denoted as $\mathbb{E} = \{E_1, E_2, \cdots, E_n\}$, at the same time. Different from other multi-expert models, the diversify loss is introduced to encourage the experts to learn to be as diverse as possible in order to gain the ability of learning to focus on different features with the same input samples. The diversity loss $L_{\text{Diversity}}$ is defined as:

$$L_{\text{Diversity}} = -\frac{1}{n-1}\sum_{i=1}^{n}\sum_{j\neq i}^{n} D_{KL}\Big(E_i(x), E_j(x)\Big), \quad (1)$$

where we compute the pairwise KL-Divergence of softmax over output logits of the sample $x$ defined as:

$$D_{KL}\left(E_i(x), E_j(x)\right) = \sum_{x\in X} E_i(x) log\frac{E_i(x)}{E_j(x)}, \quad (2)$$

where $X$ is the probability space of $E_i(x)$ and $E_j(x)$.

Finally, the overall loss for this current stage:

$$L_{\text{cascaded}}^{\text{stage 1}} = L_{\text{Cls}} + \lambda L_{\text{Diversity}}, \quad (3)$$

where in our case $L_{Cls}$ using LDAM loss [4] and $\lambda$ is a re-weighting factor.

To improve the overall efficiency of the network, an Expert Assignment Module [33] is introduced to decide the number of experts used for each sample during training or inferencing. This module provide an unique solution to address the problem without too much computational overhead. The Expert Assignment Module is trained by the routing loss which dynamically assigns and routes the experts based on the input features and predicted logits in the second stage, under the setting that all the weights of the backbone and experts are frozen. The loss function can be formulated as:

$$L_{\text{cascaded}}^{\text{stage 2}} = -\omega_p y \cdot log\Big(\frac{1}{1+e^{-y}}\Big) - \omega_n (1-y) \cdot log\Big(1 - \frac{1}{1+e^{-y}}\Big), \tag{4}$$

where the ground truth $y$ is set to 1 if and only if the prior experts fail to predict correct results and one of the later experts is able to compensate the error by giving the correct results (treated as positive samples). We then set the ground truth $y$ of all of the other cases (treated as negative samples) to be 0. And $\omega_p$ and $\omega_n$ are weights for the positive samples and negative samples respectively. The number of Expert Assignment Module will be $n-1$ eventually if we have $n$ experts in our model.

At inference and test stage, the final outputs are the arithmetic mean on logits of all assigned experts, followed by a softmax layer to produce final confidence possibilities.

### 3.3. Parallel Expert Branch

For our parallel expert branch, we train different experts based on a different training setting with focused target classification groups. As these models are trained individually and are only used for ensemble at the final decision-based voting procedure, as it makes our final predictions more robust.

We adopt the re-balancing method [13] in the training process for all of the models in this branch to handle the long-tailed distribution observed from the dataset. During each epoch in the training stage, we follow a very simple over-sampling logic by introducing a class-level repeat factor $r_c$. The repeat factor $r_c$, which controls the number of appearance of a given image $I$ belonging to class $c$, is defined as:

$$r_c = max\Big(1, \frac{t}{f_c}\Big), \tag{5}$$

where $f_c$ is the overall fraction of class $c$ over the entire training samples and $t$ is a hyper-parameter known as over-sampling threshold. Based on the definition of repeat factor, if $f_c >= t$ stands, the repeat factor will be 1 and thus no more over-sampling the images that belong to class $c$. For the cases that $f_c < t$, the images will be over-sampled every epoch. The loss function of parallel expert branch is the naive cross-entropy loss:

$$L_{\text{parallel}} = -y \cdot log\Big(\frac{1}{1+e^{-y}}\Big) - (1-y) \cdot log\Big(1 - \frac{1}{1+e^{-y}}\Big). \tag{6}$$

## 4. Experiments

### 4.1. Dataset

The NTIRE 2021 Multi-modal Aerial View Object Challenge Dataset [21] consists of two types of training images captured by Electro-Optical (EO) and Synthetic Aperture Radar (SAR) sensors over the same field of view. The EO image size is of 31 × 31 pixels while the SAR image size vary from 50 × 50 to 60 × 60 pixels. The dataset consists of a total of 10 commercial vehicle classes, including sedan, SUV, pickup truck, van, box truck, motorcycle, flatbed truck, bus, pickup truck with trailer and flatbed truck with trailer. As shown in Table 1, the number of samples of each class in the training data is extremely imbalanced with a long-tailed distribution. Based on the sample sizes of classes, the first four sample-rich classes, namely sedan, SUV, pickup truck and van are defined as the head classes. The other classes are defined as tail classes.

Table 1: Distribution of the NTIRE 2021 training data.

| Index | Type | # of Samples | % of Samples |
|---|---|---|---|
| 0 | sedan | 234,429 | 79.72% |
| 1 | SUV | 28,089 | 9.56% |
| 2 | pickup truck | 15,301 | 5.21% |
| 3 | van | 10,655 | 3.63% |
| 4 | box truck | 1,741 | 0.59% |
| 5 | motorcycle | 852 | 0.29% |
| 6 | flatbed truck | 828 | 0.28% |
| 7 | bus | 624 | 0.21% |
| 8 | pickup truck w/ trailer | 840 | 0.29% |
| 9 | flatbed truck w/ trailer | 633 | 0.22% |

In this challenge, the development-phase testing data (validation set), whose ground-truth labels are not publicly available, only the final performance scores are visible to the participants. Different from the training data, the development-phase testing data consists of 770 samples and is uniformly distributed across each class.

The testing phase testing dataset (testing set), which consists of 826 samples, is used as the final evaluation for ranking the entries. The data distribution of this split is claimed to be similar as the validation set.

### 4.2. Implementation

We use ResNet-50 [15] as the feature extractor and each expert is identical across the models being a fully connected layer with the dimension $\mathbb{R}^{2048 \times 10}$. Since the SAR image

Table 2: Development phase test results for the NTIRE2021 Multi-modal Aerial View Object Classification Challenge Track 1 with different setting and model frameworks. The ground-truth labels of the development phase test data are not yet made available to the participants, the top-1 accuracies in this table are retrieved by submitting the prediction to the validation server.

| Type | Method | Pretrained | Re-Balancing | Augmentation | Top-1 Accuracy |
|---|---|---|---|---|---|
| Single Model | Baseline (ResNet-50) | - | - | - | 15.974% |
| | | ✓ | - | - | 13.896% |
| | | - | ✓ | - | 18,182% |
| | | - | - | ✓ | 16.753% |
| | BBN [39] | - | ✓ | ✓ | 16.623% |
| | RIDE [33] | - | - | ✓ | 18.182% |
| | Cascaded Expert (Best) | - | - | ✓ | 21.039% |
| | Parallel Expert (Best) | - | ✓ | ✓ | 22.337% |
| Multiple Models | Ours | - | ✓ | ✓ | 24.675% |

sizes are not consistent in this particular dataset, we resize all samples to $124 \times 124$.

For the cascaded expert branch, we train 100 epochs for the first stage with an initial learning rate of $0.2$ and a weight decay of $0.005$. The warm-up epoch is set to $5$. The batch size is set to $512$. Then we train another 10 epochs for the second stage with a fixed learning rate of $0.01$ with a weight decay of $0.005$. Note that we do not perform the self-distillation mentioned in the original work. We have two models trained with different number of experts, namely 2 experts and 3 experts in the final submission.

For the parallel expert branch, we train for a variation of 40 to 60 epochs depending on the model with an initial learning rate of $0.01$ and a weight decay of $0.001$. The batch size is set to $64$. Since the sedan training samples significantly out-number tail-classes and even other head-classes by a very large margin. We have $4$ models trained with only the tail-classes and 3 models trained with all-classes in the final submission.

Table 3: Development phase test results for tuning the hyper-parameters for final decision-based voting fusion.

| Hyper-parameters<br>$(\alpha_{cas}, \alpha_{par}^{tail}, \alpha_{par}^{all})$ | Top-1 Accuracy |
|---|---|
| $(1, 0, 0)$ | 21.948% |
| $(0, 1, 1)$ | 23.636% |
| $(1, 1, 1)$ | 21.429% |
| $(2, 0.5, 1)$ | 24.675% |

Due to the fact that some of the models in the parallel expert branch were trained only with the tail-classes, it is natural for us to assign distinct weights during the final voting. We tuned $\alpha_{par}^{tail}$, $\alpha_{par}^{all}$ and $\alpha_{cas}$ base on the validation performance from the server. As the submission quotas were limited during the time of competition, we only try different combinations of the hyper-parameters from $[0, 0.5, 1.0, 2.0]$ listed in Table 3 which also includes the results of using only cascaded expert branch or only parallel expert branch. The final fusion weights are tuned as $\alpha_{cas} = 2$, $\alpha_{par}^{tail} = 0.5$, and $\alpha_{par}^{all} = 1$.

Our experimental environment is Python 3.7, Pytorch 1.4 and mmclassification 0.8.0 with one Nvidia Quadro GV100.

### 4.3. Experiment Results

Table 2 shows the experimental results on development phase testing dataset (validation set), which includes experiments from pretrained weights, augmentation and re-balancing strategy on the baseline model ResNet-50. We also conduct additional experiments using the original settings of BBN [39] and RIDE [33]. Since more than one model are trained in both cascaded expert branch and parallel expert branch, the best single model from each of the branch is recorded.

Table 4: Latest update of the test results for the NTIRE2021 Multi-modal Aerial View Object Classification Challenge Track 1 (SAR).

| Team | Top-1 Accuracy<br>(# of Correct Prediction) |
|---|---|
| Team A | 34.615% (286) |
| Team B | 26.634% (220) |
| Team C | 26.392% (218) |
| Team D | 25.061% (207) |
| Ours | 26.029% (215) |

The top 5 teams from the latest update of the test results for the NTIRE2021 Multi-modal Aerial View Object Classification Challenge Track 1 (SAR) are listed in terms of the top-1 accuracy in the testing phase in Table 4. With only 826 testing samples, the top ranking teams actually have very similar performance with only a couple samples difference which shows that our multi-expert architecture have the potential to address this problem.

### 4.4. Ablation Study

Augmentation is a common technique widely used in the deep learning task to diversify the distribution of the original dataset to prevent over-fitting. However, this is not the case when we use SAR image as input to the CNN models as the SAR image coordinates range and azimuth cannot be arbitrarily coordinated like other aerial view imagery. This observation is also mentioned in the [40], suggesting that we may come up with an augmented sample, that can never be generated by the SAR, by randomly rotating or cropping the original samples. To understand the effect of this, we present some experiments on whether including the data augmentation in our training process, as shown in Table 5 where we record the best top-1 accuracy that can be achieved after trying different parameters for the augmentation.

Table 5: Ablation studies on the development phase test results for the NTIRE2021 Multi-modal Aerial View Object Classification Challenge Track1.

| Augmentation Method | Top-1 Accuracy |
|---|---|
| Baseline (ResNet-50) | 15.974% |
| Random Cropping | 12.987% |
| Center Cropping | 12.338% |
| Rotation | 15.584% |
| Flipping | 16.494% |
| Rotation + Flipping | 16.753% |

From our ablation experimental results about the effect of augmentation on SAR image, it can be concluded that most of the conventional augmentation methods fail to enhance the performance of the SAR classification when using the CNN structure. However, due to the limited size or distribution of the dataset, we still adopt rotation and flipping in all of our training pipeline as it gives a small margin of improvement.

## 5. Conclusion

In this paper, we present a two-branch multi-expert framework, consisting of a cascaded expert branch and a parallel expert branch, which aims to tackle the atypical long-tailed distribution observed in the NTIRE 2021 Multi-modal Aerial View Object Challenge Dataset. Our proposed model, named Cascading and Parallel Experts, achieves 26.029% on the testing set in Track 1 using only low-resolution SAR images as input. The experimental results show the promise of our proposed method, which can possess the effectiveness as it achieves top-tier performance among 157 participants for Aerial View Object Classification using only Synthetic Aperture Radar imagery.

## References

[1] Moussa Amrani and Feng Jiang. Deep feature extraction and combination for synthetic aperture radar target classification. *Journal of Applied Remote Sensing*, 11(4):042616, 2017. 1

[2] Nantheera Anantrasirichai, Juliet Biggs, Fabien Albino, P Hill, and D Bull. Application of machine learning to classification of volcanic deformation in routinely generated insar data. *Journal of Geophysical Research: Solid Earth*, 123(8):6592–6606, 2018. 2

[3] Michael Lee Bryant and Frederick D Garber. Svm classifier applied to the mstar public data set. In *Algorithms for Synthetic Aperture Radar Imagery VI*, volume 3721, pages 355–360. International Society for Optics and Photonics, 1999. 2

[4] Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. *arXiv preprint arXiv:1906.07413*, 2019. 2, 3

[5] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority oversampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002. 2

[6] Sizhe Chen and Haipeng Wang. Sar target recognition based on deep learning. In *2014 International Conference on Data Science and Advanced Analytics (DSAA)*, pages 541–547. IEEE, 2014. 2

[7] Jun Hoo Cho and Chan Gook Park. Multiple feature aggregation using convolutional neural networks for sar image-based automatic target recognition. *IEEE Geoscience and Remote Sensing Letters*, 15(12):1882–1886, 2018. 2

[8] Hsin-Ping Chou, Shih-Chieh Chang, Jia-Yu Pan, Wei Wei, and Da-Cheng Juan. Remix: Rebalanced mixup. In *European Conference on Computer Vision*, pages 95–110. Springer, 2020. 2

[9] Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9268–9277, 2019. 2

[10] Sheng Deng, Lan Du, Chen Li, Jun Ding, and Hongwei Liu. Sar automatic target recognition based on euclidean distance restricted autoencoder. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(7):3323–3333, 2017. 2

[11] Ganggang Dong, Gangyao Kuang, Na Wang, Lingjun Zhao, and Jun Lu. Sar target recognition via joint sparse representation of monogenic signal. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(7):3316–3328, 2015. 2

[12] Mikel Galar, Alberto Fernández, Edurne Barrenechea, and Francisco Herrera. Eusboost: Enhancing ensembles for highly imbalanced data-sets by evolutionary undersampling. *Pattern recognition*, 46(12):3460–3471, 2013. 2

[13] Agrim Gupta, Piotr Dollár, and Ross B. Girshick. LVIS: A dataset for large vocabulary instance segmentation. *CoRR*, abs/1908.03195, 2019. 4

[14] Hui Han, Wen-Yuan Wang, and Bing-Huan Mao. Borderline-smote: a new over-sampling method in imbalanced data sets learning. In *International conference on intelligent computing*, pages 878–887. Springer, 2005. 2

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2, 4

[16] Lloyd H Hughes, Michael Schmitt, Lichao Mou, Yuanyuan Wang, and Xiao Xiang Zhu. Identifying corresponding patches in sar and optical images with a pseudo-siamese cnn. *IEEE Geoscience and Remote Sensing Letters*, 15(5):784–788, 2018. 2

[17] Lizhong Jin, Junjie Chen, and Xinguang Peng. Joint classification of complementary features based on multitask compressive sensing with application to synthetic aperture radar automatic target recognition. *Journal of Electronic Imaging*, 27(5):053034, 2018. 1

[18] Yu Junfei, Li Jingwen, Sun Bing, and Jiang Yuming. Barrage jamming detection and classification based on convolutional neural network for synthetic aperture radar. In *IGARSS 2018-2018 IEEE International Geoscience and Remote Sensing Symposium*, pages 4583–4586. IEEE, 2018. 1

[19] Bingyi Kang, Saining Xie, Marcus Rohrbach, Zhicheng Yan, Albert Gordo, Jiashi Feng, and Yannis Kalantidis. Decoupling representation and classifier for long-tailed recognition. *arXiv preprint arXiv:1910.09217*, 2019. 2

[20] Jaehyung Kim, Jongheon Jeong, and Jinwoo Shin. M2m: Imbalanced classification via major-to-minor translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13896–13905, 2020. 2

[21] Jerrick Liu, Oliver Nina, Radu Timofte, et al. NTIRE 2021 multi-modal aerial view object classification challenge. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2021. 2, 4

[22] Xu-Ying Liu, Jianxin Wu, and Zhi-Hua Zhou. Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 39(2):539–550, 2008. 2

[23] Jie Mei, Jenq-Neng Hwang, Suzanne Romain, Craig Rose, Braden Moore, and Kelsey Magrane. Video-based hierarchical species classification for longline fishing monitoring. *arXiv preprint arXiv:2102.03520*, 2021. 2

[24] Rui Min, Hai Lan, Zongjie Cao, and Zongyong Cui. A gradually distilled cnn for sar target recognition. *IEEE Access*, 7:42190–42200, 2019. 2

[25] Hien M Nguyen, Eric W Cooper, and Katsuari Kamei. Borderline over-sampling for imbalanced data classification. *International Journal of Knowledge Engineering and Soft Data Paradigms*, 3(1):4–21, 2011. 2

[26] Hemani Parikh, Samir Patel, and Vibha Patel. Classification of sar and polsar images using deep learning: a review. *International Journal of Image and Data Fusion*, 11(1):1–32, 2020. 2

[27] Andrew Profeta, Andres Rodriguez, and H Scott Clouse. Convolutional neural networks for synthetic aperture radar classification. In *Algorithms for synthetic aperture radar imagery XXIII*, volume 9843, page 98430M. International Society for Optics and Photonics, 2016. 1

[28] Ronghua Shang, Jianghai He, Jiaming Wang, Kaiming Xu, Licheng Jiao, and Rustam Stolkin. Dense connection and depthwise separable convolution based cnn for polarimetric sar image classification. *Knowledge-Based Systems*, 194:105542, 2020. 2

[29] Jiaqi Shao, Changwen Qu, Jianwei Li, and Shujuan Peng. A lightweight convolutional neural network based on visual attention for sar image target classification. *Sensors*, 18(9):3039, 2018. 2

[30] Umamahesh Srinivas, Vishal Monga, and Raghu G Raj. Sar automatic target recognition using discriminative graphical models. *IEEE transactions on aerospace and electronic systems*, 50(1):591–606, 2014. 2

[31] L Wang, KA Scott, L Xu, and D Clausi. J. ice concentration estimation from dual-polarized sar images using deep convolutional neural networks. *IEEE (Transactions on Geoscience and Remote Sensing)*, 2014. 2

[32] Puyang Wang, He Zhang, and Vishal M Patel. Sar image despeckling using a convolutional neural network. *IEEE Signal Processing Letters*, 24(12):1763–1767, 2017. 2

[33] Xudong Wang, Long Lian, Zhongqi Miao, Ziwei Liu, and Stella X Yu. Long-tailed recognition by routing diverse distribution-aware experts. *arXiv preprint arXiv:2010.01809*, 2020. 2, 3, 4, 5

[34] Xudong Wang, Long Lian, Zhongqi Miao, Ziwei Liu, and Stella X Yu. Long-tailed recognition by routing diverse distribution-aware experts. *arXiv preprint arXiv:2010.01809*, 2020. 3

[35] Liuyu Xiang, Guiguang Ding, and Jungong Han. Learning from multiple experts: Self-paced knowledge distillation for long-tailed classification. In *European Conference on Computer Vision*, pages 247–263. Springer, 2020. 3

[36] Shuyuan Yang, Min Wang, Zhixi Feng, Zhi Liu, and Rundong Li. Deep sparse tensor filtering network for synthetic aperture radar images classification. *IEEE transactions on neural networks and learning systems*, 29(8):3919–3924, 2018. 1

[37] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017. 2

[38] Qun Zhao and Jose C Principe. Support vector machines for sar automatic target recognition. *IEEE Transactions on Aerospace and Electronic Systems*, 37(2):643–654, 2001. 2

[39] Boyan Zhou, Quan Cui, Xiu-Shen Wei, and Zhao-Min Chen. Bbn: Bilateral-branch network with cumulative learning for long-tailed visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9719–9728, 2020. 2, 5

[40] Xiao Xiang Zhu, Sina Montazeri, Mohsin Ali, Yuansheng Hua, Yuanyuan Wang, Lichao Mou, Yilei Shi, Feng Xu, and Richard Bamler. Deep learning meets sar, 2021. 6

[41] Xiao Xiang Zhu, Devis Tuia, Lichao Mou, Gui-Song Xia, Liangpei Zhang, Feng Xu, and Friedrich Fraundorfer. Deep learning in remote sensing: A comprehensive review and list of resources. *IEEE Geoscience and Remote Sensing Magazine*, 5(4):8–36, 2017. 2