applied sciences

MDPI

Article

An Adaptive Weighted Method for Remote Sensing Image Retrieval with Noisy Labels

Xueqing Tian [1], Dongyang Hou [1,*], Siyuan Wang [2], Xuanyou Liu [1] and Huaqiao Xing [3]

[1] School of Geosciences and Info-Physics, Central South University, Changsha 410083, China; 215011043@csu.edu.cn (X.T.); 215012160@csu.edu.cn (X.L.)
[2] State Key Laboratory of Information Engineering in Surveying, Mapping, and Remote Sensing, Wuhan University, Wuhan 430079, China; wsy1998@whu.edu.cn
[3] School of Surveying and Geo-Informatics, Shandong Jianzhu University, Jinan 250101, China; xinghuaqiao18@sdjzu.edu.cn
* Correspondence: houdongyang1986@csu.edu.cn

Abstract: Due to issues with sample quality, there is an increasing interest in deep learning models that can handle noisy labels. Currently, the optimal way to deal with noisy labels is by combining robust active and passive loss functions. However, the weighting parameters for these functions are typically determined manually or through a large number of experimental iterations, and even the weighting parameters change as the dataset and the noisy rate change. This can lead to suboptimal results and be time-consuming. Therefore, we propose an adaptively weighted method for the combined active passive loss (APL) in remote sensing image retrieval with noisy labels. First, two metrics are selected to measure the noisy samples: the ratio of the entropy to the standard deviation and the difference of the predicted probabilities. Then, an adaptive weighted learning network with a hidden layer is designed to dynamically learn the weighting parameters. The network takes the above two metrics as inputs and is trained concurrently with the feature extraction network in each batch, without significantly increasing the computational complexity. Extensive experiments demonstrate that our improved APL method outperforms the original manually weighted APL method and other state-of-the-art robust loss methods while saving the time on manual parameter selection.

Keywords: noisy labels; robust loss; remote sensing image retrieval; deep learning

check for updates

Citation: Tian, X.; Hou, D.; Wang, S.; Liu, X.; Xing, H. An Adaptive Weighted Method for Remote Sensing Image Retrieval with Noisy Labels. *Appl. Sci.* **2024**, *14*, 1756. https://doi.org/10.3390/app14051756

Academic Editor: Atsushi Mase

Received: 25 January 2024
Revised: 19 February 2024
Accepted: 20 February 2024
Published: 21 February 2024

1. Introduction

The demand for the fast and efficient retrieval of images from large remote sensing databases has become increasingly urgent due to military and civilian needs in geospatial information science [1]. Content-based remote sensing image retrieval (CBRSIR) is an effective method to solve meet this demand [2] and has attracted an increasing amount of research interest.

In recent years, various methods based on deep learning models, which can automatically learn the high-level semantic features of remote sensing images, have become the mainstream method for CBRSIR [3–5]. These deep learning model-based methods are data-driven and require large amounts of sample data. Therefore, to reduce the cost of labelling large datasets, many researchers have proposed clustering, semi-automatic labelling and crowdsourcing methods in real-world application scenarios [6–8]. However, these methods can introduce noisy labels into the sample datasets. According to the literature [9], existing datasets contain between 8.0% and 38.5% noisy labels. Such noisy labels may lead to overfitting of the deep learning-based methods [10], which can reduce the performance of remote sensing image retrieval. For example, Li et al. [11] found that noisy labels significantly affect the accuracy of deep learning-based classifiers, which can also negatively impact retrieval results. Kang et al. [12] also confirmed that deep learning models are not robust enough in benchmarking datasets with noisy labels. Therefore, it is crucial to consider the effect of noisy labels during deep learning model training.

To reduce the effect of noisy labels, several robust loss functions have been proposed. For example, Ghosh et al. [13] proposed a mean absolute error (MAE) loss, which is more robust than cross-entropy (CE). However, it converges slowly due to gradient saturation. By combining the MAE with the faster converging CE, Zhang et al. [14] proposed a generalised cross-entropy (GCE). Inspired by the symmetry of Kullback–Leibler divergence, a robust symmetric cross-entropy (SCE) [15] is proposed by combining the robust reverse cross-entropy (RCE) and CE with two weighting hyperparameters. Additionally, Chen et al. [16] proposed an adaptive cross-entropy (ACE) by replacing the two weighting parameters of the SCE with the output probability of the deep learning network. This approach can eliminate the need to manually select the weighting parameters for the SCE. Furthermore, Ma et al. [17] categorised existing robust losses into active and passive losses and introduced an active passive loss (APL) by combining the two different types of robust losses with two weighting hyperparameters. This method provides a theoretical explanation for the effectiveness of the combination losses in combating noisy labels. Compared with other robust methods, the APL method has the best results. However, the APL method is time-consuming, and it may not always achieve optimal performance, as the two weighting parameters are typically determined manually or through a large number of experimental iterations. More unfortunately, the two weighting parameters vary with the dataset and the noise rate. This limitation restricts its widespread use in real-world applications, especially in contexts where noise levels in datasets are unknown. In addition, the ACE parameter replacement method is not applicable to APL.

To solve this problem, we propose an adaptive weighted method for the robust APL method in remote sensing image retrieval. Specifically, a new metric based on the entropy and standard deviation of the predicted probabilities of the samples is developed to represent the complexity of the sample. Additionally, the predicted probability difference is chosen as a second metric of whether the sample has a noisy label. Then, an adaptive weighted learning network (AWNet) with one hidden layer is designed to dynamically learn the weighting parameters in each training batch using the above two metrics as inputs. Our code is available at https://github.com/GeoRSAI/APL_AWNet (accessed on 24 January 2024).

The rest of this paper is organised as follows. Section 2 reviews related works on CBRSIR based on deep learning, robust loss functions and multilayer perceptron for remote sensing. Section 3 details our proposed method, including the framework, two metrics and AWNet. The experimental results and analysis are presented in Section 4, while Section 5 provides the conclusion of this paper.

2. Related Works

2.1. CBRSIR Based on Deep Learning

Deep learning has been extensively applied in CBRSIR and has achieved excellent performance. It has gradually replaced the low- and mid-level feature-based methods. This is due to the ability of deep neural networks to extract high-level semantic features, which can better represent the content of remote sensing images. For instance, Zhou et al. [18] trained a mainstream convolutional neural network model on a remote sensing dataset, and the performance of the model was significantly better than that of the low- and mid-level feature-based methods. Other research has attempted to enhance feature extraction by using more complex network structures, such as the contrastive self-supervised learning network [19] and attention mechanisms [20,21]. On clean datasets, many of the current methods have achieved near-saturation retrieval accuracy. However, when the training data contains noisy labels, the model tends to overfit to noisy samples. This significantly reduces the accuracy of the classification model and, subsequently, the retrieval accuracy. To solve the above problem, Li et al. [22] proposed a fault-tolerant deep learning method for remote sensing scene classification. The method utilises ensemble learning to enhance the accuracy of the error correction of noisy labels, with data cleaning as the underlying concept. However, the method fuses several large networks and iterates many times in the training process, so the number of parameters and computations is large. Damodaran et al. [23] proposed a loss with entropic optimal transport (CLEOT), which designed a robust loss by exploring the joint publication of images and labels, and achieved good results, but the method performed poorly at low proportion noise. Overall, there is a need to complement the research on remote sensing image retrieval with noisy labels.

2.2. Noise Robust Loss Functions

A loss function is considered noise-robust if the classifier achieves the same classification accuracy on both noisy and noise-free data [24]. Compared to other robust methods such as relabelling [25] and sample importance weighting [26], robust loss is a simpler and more general method. Currently, symmetric loss is one of the mainstream robust loss functions [27,28], such as MAE [13], RCE [15] and so on. To make any loss symmetric, Ma et al. [17] proposed normalised loss functions using a simple normalisation operation. However, this operation actually changes the form of loss functions, which leads to difficulties in optimisation. Consequently, the fitting ability of the symmetric loss function is limited by the symmetry condition [29], and the model underfitting problem is prone to occur. Inspired by the advantages of symmetric [15] or complementary learning [30], the APL [17] framework was proposed for robust and sufficient learning, which is combined with active loss and passive loss to mutually reinforces each other and solve the model underfitting problem. Although APL has shown excellent performance among many robust loss functions, it has two hyperparameters (α and β), which cannot maintain optimal performance on any dataset without tuning. Recently, asymmetric loss functions have been proposed as a new class of robust loss functions by Zhou et al. [31]. Meanwhile, several asymmetric loss functions such as asymmetric unhinged loss (AUL) and asymmetric exponential loss (AEL) have also been proposed by Zhou et al. [31]. However, this type of asymmetric loss function requires that each sample in the training dataset has a higher probability of being labelled with a true semantic label than any other class of labels during image classification. This means that it is ineffective for noisy label types that are easily confused. Therefore, in this paper, the active passive loss function (APL) is still chosen as an improved benchmark for remote sensing image retrieval with noisy labels.

2.3. Multilayer Perceptron for Remote Sensing

As a representative neural network structure, multilayer perceptron (MLP) is widely used in remote sensing tasks such as remote sensing image classification [32–34], object detection [35] and change detection [36,37]. The MLP can have multiple hidden layers between the input and output layers, but the simplest MLP has only one hidden layer. Despite its simple structure, numerous computer vision experiments [38] have demonstrated that the MLP still has the same feature representation capabilities as traditional convolution and transformers, even under complex network and large dataset conditions. In addition, the MLP has excellent compatibility with CNNs [39]. For example, by combining the spectral features extracted by MLP with the spatial features represented by a CNN, Zhang et al. [40] used a rule-based decision fusion method to integrate the MLP with the CNN and achieved excellent classification accuracy of very fine spatial resolution remote sensing images. It is also inspired by meta-learning [41] and makes it possible to use MLP for the automatic determination of hyperparameters such as weights. Therefore, we adopt the MLP as the main structure of our AWNet to automatically acquire active and passive loss weights, which, in turn, improves its generalisation.

3. Methodology

In this section, we first introduce the APL and analyse its limitations. Then, we present the framework of our method, which consists of feature extraction, AWNet and image retrieval components. Next, we explain the meaning and role of the two metrics required for the adaptive determination of the APL weights, namely the ratio of entropy to standard deviation (abbreviated as RES) and δ. Finally, we introduce our core method of the AWNet and describe the algorithm in detail, showing the connection between the AWNet, APL weights and classification.

3.1. The Active Passive Loss (APL)

To address the model underfitting problem caused by noisy labels, Ma et al. [17] proposed a framework to construct robust loss functions called APL, which combines the active loss (i.e., CE, normalised CE, focal loss and normalised focal loss) and the passive loss (i.e., MAE, normalised MAE, RCE and normalised RCE). The APL l_{APL} can be defined as follows.

$$l_{APL}= \alpha \cdot l_{Active} + \beta \cdot l_{Passive} \tag{1}$$

where variables α and β are used to compensate for robust losses and $\alpha, \beta > 0$. The term l_{Active} represents active losses that explicitly maximise the network's output probability at the class position specified by the label. The term $l_{Passive}$ represents passive losses, which explicitly minimise the probability at least one other class position. Therefore, for noisy samples, more passive learning can preserve the effective information of the samples and avoid misguiding the model. Conversely, more active learning can speed up the learning of the model and avoid underfitting. However, we found that, for different datasets, the weights of active and passive losses need to be manually adjusted to achieve the best performance, which increases the training cost and limits the generalisation performance of the model. The aim of our method is to automatically adjust the weighting hyperparameters α and β in remote sensing image retrieval.

3.2. Framework of Our Method

The framework of our method is shown in Figure 1. Our method mainly consists of feature extraction module, the AWNet module and a querying module.

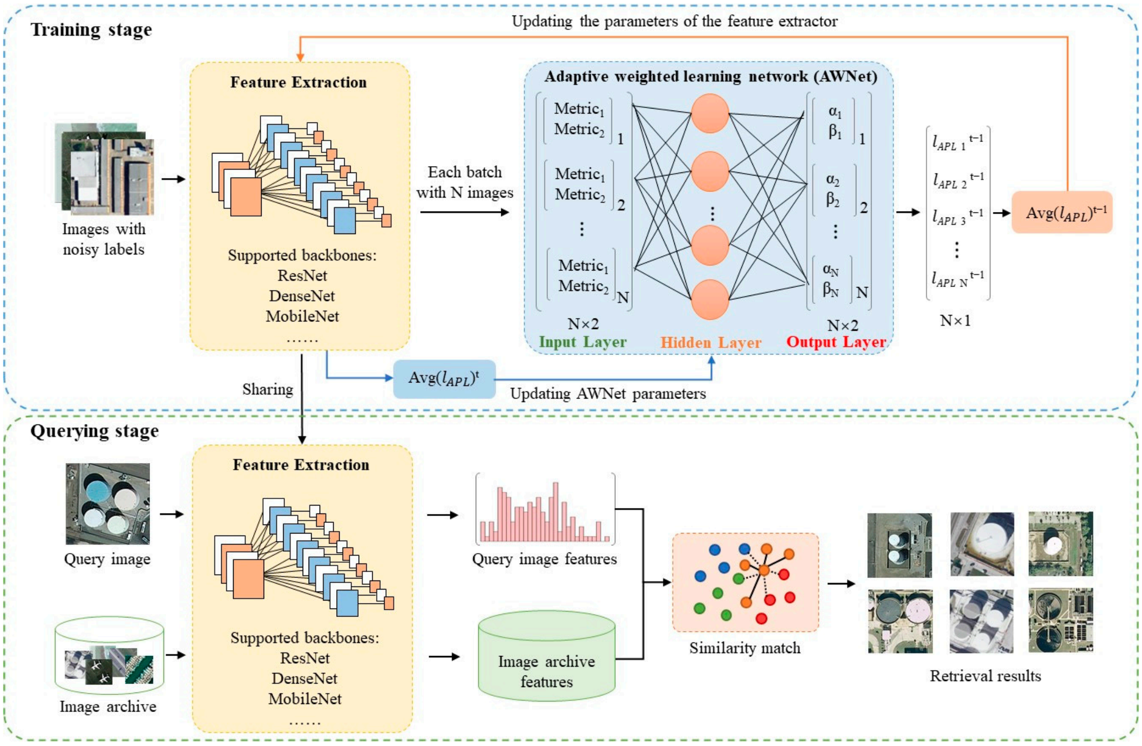

Figure 1. Framework of our proposed method.

The feature extraction module can use any type of convolutional neural network model (e.g., ResNet, DenseNet, MobileNet, etc.) with the APL method to learn image features from training samples with noisy labels. The AWNet module aims to dynamically adjust the weighting hyperparameters of the APL method according to the predicted probability of the feature extraction module. In the training stage, two different metrics for the complexity and noise level of each sample image in a batch can be computed based on the predicted probability of the feature extraction module. The metric values of all images within the batch are used together as inputs to the AWNet module. With the AWNet, a different set of weighting hyperparameters, α and β, is obtained for each sampled image. To mitigate the negative effects of noisy labels in training samples, it is recommended to use more passive losses, which require a large β. Conversely, for clean training samples with correct labels, it is advisable to use more active losses to facilitate quick convergence of the model, which requires a large α. The active passive loss can then be calculated for each image. The average APL of all images within a batch is used to update the classifier parameters of the image feature extraction module. In turn, the average value of the APL within the same batch is recalculated using the newly updated classifier and used to update the AWNet. Details of the two metrics and the adaptive weighted learning network are described in the following section.

In the query stage, we first use the optimal model obtained in the training stage to extract the features of the query image. Then, we calculate the similarity between the feature vector of the query image and the feature vector of each image in the database. In this paper, the Euclidean distance is selected as the similarity metric. Finally, based on the descending order of image feature similarity, the top-ranked images are selected as the final retrieval results.

3.3. Two Metrics

The weighting hyperparameters of the combined robust loss are closely related to the training sample. For example, complex datasets require more active learning (such as large α) and less passive learning (such as small β) to achieve good performance [17]. Furthermore, the weighting parameters are correlated with the noise rate in the trained samples, because noisy samples require more passive learning. Therefore, two metrics are designed to reflect the sample complexity and noise level.

In the field of digital image processing, the complexity of an image is reflected by its information entropy $E(.)$ which is defined as Equation (2). Furthermore, Equation (3) defines the standard deviation of the output layer of the classifier $S(.)$. This indirectly reflects the complexity of the sample, because a more complex image is more difficult for the classifier, resulting in a greater difference in the predicted probability of each category in the output layer and a lower value of $S(.)$.

$$E(p_i) = -\sum_{i=1}^{C} (p_i log p_i) \tag{2}$$

$$S(p_i) = \sqrt{\frac{1}{C}\sum_{i=1}^{C} (\hat{p}_i - p_i)^2} \tag{3}$$

where C represents the total number of labels, p_i represents the predicted probability of each class from the output layer of the classifier, and $\hat{p}_i$ means the average value of p_i.

It is important to note that the value of $E(.)$ increases and the value of $S(.)$ decreases as the image complexity increases. Therefore, as shown in Equation (4), the first metric is defined as the ratio of entropy to standard deviation (abbreviated as RES). The higher the RES, the higher the complexity of the sample.

$$\text{RES} = \frac{E(p_i)}{S(p_i)} \tag{4}$$

In addition, the probability difference δ [42] is selected as the second metric, which can be used to determine the noise level of the sample. It is defined as follows:

$$\delta = p_y - p_n \tag{5}$$

where p_y denotes the probability of being predicted as a correct category by the classifier, p_n denotes the maximum probability of being predicted as other incorrect categories, and $\delta\epsilon[-1,\ 1]$. The smaller the δ value of a sample when δ < 0, the more likely it is to be a noisy sample.

3.4. Adaptive Weighted Learning Network

Inspired by meta-learning [41], we expect the model to be able to learn the relationship between training samples and weighting hyperparameters automatically. At the same time, we aim to achieve this without incurring high computational costs. To accomplish this, we construct an adaptive weight learning network (AWNet) based on a multilayer perceptron. The network consists of an input layer, a hidden layer (with 100 neurons) and an output layer. The hidden and output layers consist of a linear model and an activation function. To avoid the issues of gradient explosion and gradient disappearance, ReLU converts the input gradient into 1 and 0. Therefore, the activation function of the hidden layer is ReLU, and the activation function of the output layer is Sigmoid. Full connectivity is used between the different layers. Specifically, the complexity and noise level of the training samples are expressed quantitatively by two metrics: RES and δ. The two metrics are then used as the input of AWNet to fit their relationship with α and β, which can automatically weigh each sample. The forward computation procedure of AWNet can be written as

$$(\alpha,\ \beta) = \mu\left(W_2^t\left(\mathrm{ReLU}\left(W_1^t(\mathrm{RES}, \delta)\right)\right.\right. \tag{6}$$

where μ and ReLU are the Sigmoid and ReLU activation functions, respectively. $W_1^T$and W_2^T are the feature weights between the input layer and the hidden layer and between the hidden layer and the output layer, respectively.

Algorithm 1 describes the process for updating parameters of our proposed AWNet and the classifier during the training stage. To ensure the reliability of the classifier's output probability, we employ a fixed-weight APL (α = 1, β = 1) for pretraining the classifier in the first t_p epochs. In our experiments, we set the number of pretraining epochs t_p to 3. Detailed information about the setting of t_p will be introduced in Section 4.5. Once the pretraining is completed, the classifier proceeds to the formal training stage. In this stage, we calculate the RES and δ for each sample by using the output probabilities of the classifier. These two metrics are then input into AWNet to obtain α and β for each sample. Subsequently, we update the parameters of the classifier by Equation (1) and then input the same samples into the updated classifier to update the AWNet parameters with the new loss. The classifier and AWNet parameters are iteratively updated until training is complete. Our method can be trained directly on noisy data D_n without the need for additional clean data as a guide.

Algorithm 1: Training Process of AWNet and classifier

```
Input:
Data: training dataset with noisy labels $D_n$;
Component: classifier $f(\cdot)$ and AWNet $\mathbf{w}(\cdot)$;
Parameter: $\alpha$, $\beta$ pretraining epochs $t_p$, max epochs $t_{max}$ and iteration per-epoch $e$;
Output: Well-trained $f(\cdot)$ and $\mathbf{w}(\cdot)$.
$i = 1$;
while $i < t_{max} + 1$ do:
    if $i < t_p + 1$, then:
        $j = 1$, $\alpha = 1$, $\beta = 1$;
        while $j < e + 1$ do:
            Train the classifier $f(\cdot)$ by $D_n$;
            Update $f(\cdot)$ according to Equation (1);
            $j = j + 1$
        end while
    else:
        $k = 1$
        while $k < e + 1$ do:
            Train the classifier $f(\cdot)$ by $D_n$;
            Calculate RES by Equation (4) and $\delta$ by Equation (5);
            Get $\alpha$ and $\beta$ by Equation (6)
            Update $f(\cdot)$ according to (1);
            Train the classifier $f(\cdot)$ by $D_n$;
            Update $\mathbf{w}(\cdot)$ according to (1);
            $k = k + 1$
        end while
    $i = i + 1$
end while
```

4. Experiments and Analysis

4.1. Datasets and Experimental Setup

Three widely used public remote sensing image datasets: UC-Merced dataset (UCMD) [43], aerial image dataset (AID) [44] and Northwestern Polytechnical University dataset (NWPU) [45] are used to evaluate the effectiveness of our proposed method. The UCMD dataset [43] consists of 21 categories, and each category contains 100 images with a size of 256 × 256 pixels. The AID dataset [44] contains 30 categories and a total of 10,000 images, with 220–420 images per category. The image size is 600 × 600 pixels. The NWPU dataset [45] contains 31,500 images and 45 categories. There are 700 images per category, and each image is 256 × 256 pixels in size. These three datasets can be used to validate the robustness of the method, as they have different levels of complexity and intra-class diversity. Examples of all categories in the three datasets are shown in Figures 2–4.

In the experiments, we randomly select 60% of the images as the training set, 20% of the images as the validation set and 20% of the images as the test set. To evaluate the effectiveness of the methods against noisy labels, we add different proportions of symmetric noise (e.g., 5%, 10%, 20% and 30%) to simulate noisy labels in the training and validation sets. Specifically, the label is flipped uniformly across all the classes with probability p, regardless of the similarity between the classes. In this case, the label transition matrix E has the entries $1 - p$ in the diagonal and p in the off-diagonal elements.

To comprehensively evaluate the effectiveness of our method, we compare it with manual weighting methods on different backbone models, including ResNet [46], DenseNet [47] and MobileNet [48]. The compared methods are MAE [13]; GCE [14]; SCE [15]; RCE [15]; ACE [16]; AUL [31]; AEL [28] and four APL combinations, namely αNCE (normalised cross-entropy loss) + βMAE, αNCE + βRCE, αNFL (normalised focal loss) + βMAE and αNFL + βMAE with 12 different combinations of the weighting hyperparameters α and β [17]. These 12 weights are recommended in the literature [17]. In addition, mAP (mean average precision) is used as a metric to evaluate the retrieval performance.

All experiments are repeated three times to ensure the reliability of the results. All methods are performed with ImageNet pretrained classifiers and the Adam optimiser. The initial learning rate and weight decay are both set to 0.00015. To avoid unreliable noisy sample evaluation results at the beginning of training, α and β are set to 1 for the first three epochs. Our method's feature extraction module uses ResNet50 as the basic backbone. AWNet also uses Adam as the optimiser, with an initial learning rate of 0.001 and a weight decay of 0.0001. All models are trained using PyTorch 1.8.2 on a single NVIDIA GeForce RTX 3090 GPU with a batch size of 128 for 20 epochs.

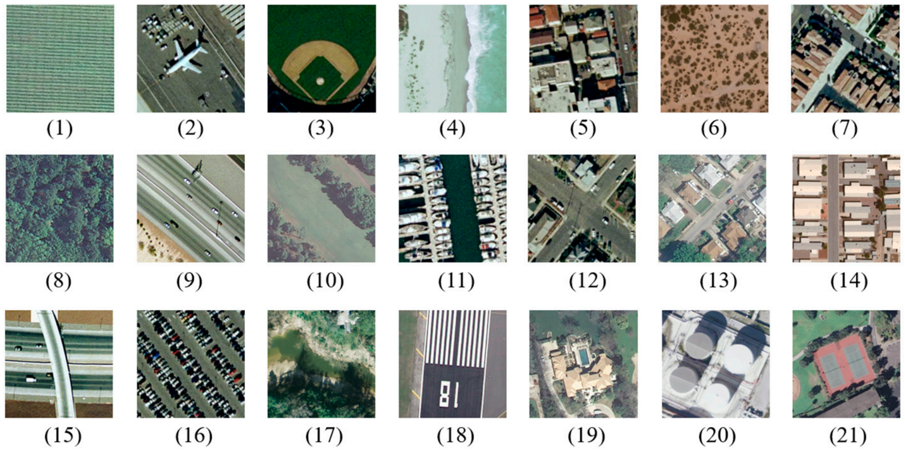

Figure 2. Examples of the UCMD dataset: (1) agricultural; (2) airplane; (3) baseball diamond; (4) beach; (5) buildings; (6) chaparral; (7) dense residential; (8) forest; (9) freeway; (10) golf course; (11) harbour; (12) intersection; (13) medium density residential; (14) mobile home park; (15) overpass; (16) parking lot; (17) river; (18) runway; (19) sparse residential; (20) storage tanks; (21) tennis courts.

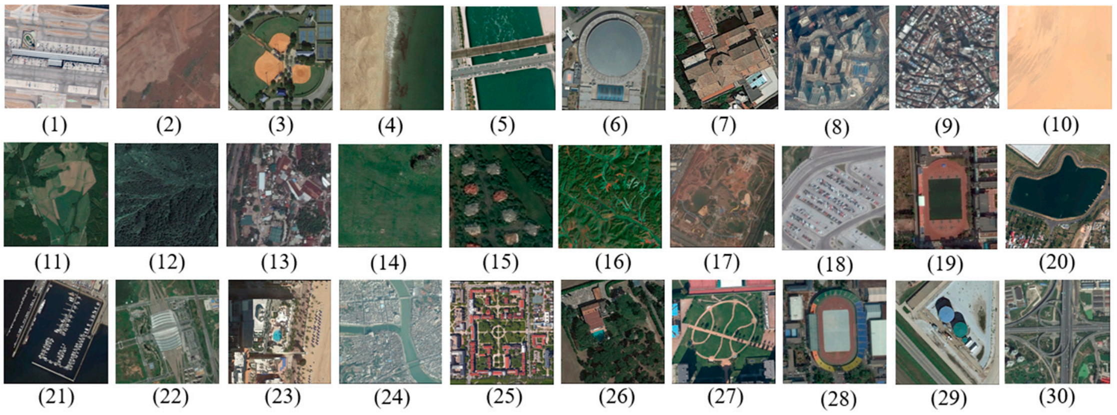

Figure 3. Examples of the AID dataset: (1) airport; (2) bare land; (3) baseball field; (4) beach; (5) bridge; (6) centre; (7) church; (8) commercial; (9) dense residential; (10) desert; (11) farmland; (12) forest; (13) industrial; (14) meadow; (15) medium residential; (16) mountain; (17) park; (18) parking; (19) playground; (20) pond; (21) port; (22) railway station; (23) resort; (24) river; (25) school; (26) sparse residential; (27) square; (28) stadium; (29) storage tanks; (30) viaduct.

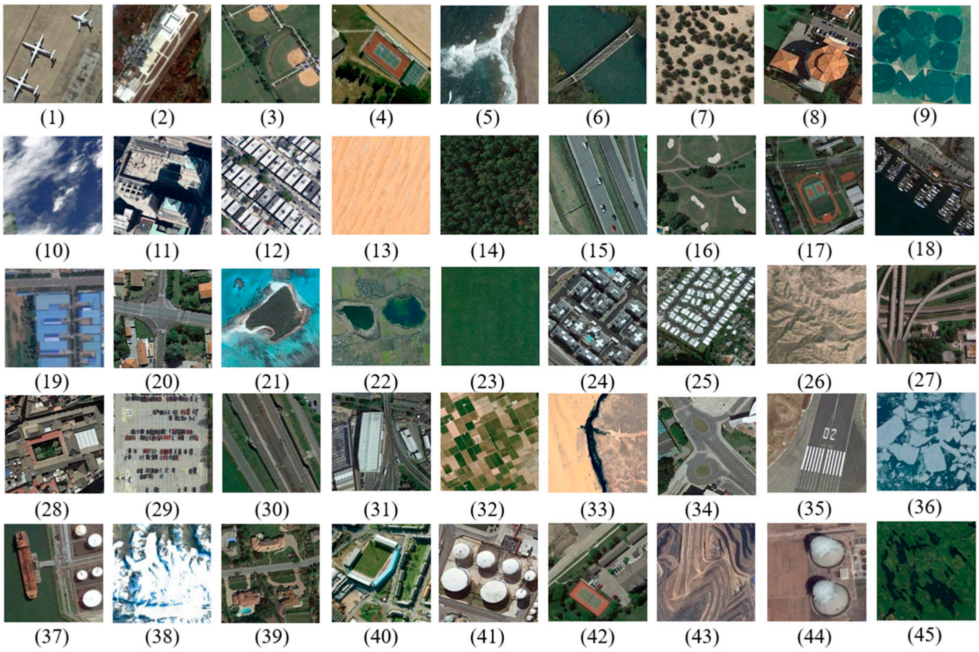

Figure 4. Examples of the NWPU dataset: (1) airplane; (2) airport; (3) baseball diamond; (4) basketball court; (5) beach; (6) bridge; (7) chaparral; (8) church; (9) circular farmland; (10) cloud; (11) commercial area; (12) dense residential; (13) desert; (14) forest; (15) freeway; (16) golf course; (17) ground track field; (18) harbour; (19) industrial area; (20) intersection; (21) island; (22) lake; (23) meadow; (24) medium residential; (25) mobile home park; (26) mountain; (27) overpass; (28) palace; (29) parking lot; (30) railway; (31) railway station; (32) rectangular farmland; (33) river; (34) roundabout; (35) runway; (36) sea ice; (37) ship; (38) snowberg; (39) sparse residential; (40) stadium; (41) storage tank; (42) tennis court; (43) terrace; (44) thermal power station; (45) wetland.

4.2. Experiments on Adaptive Weights versus Manual Weights

In these experiments, we chose α NCE + β RCE as a representative of the APL. Tables 1 and 2 display the retrieval performance of the CE, APL with 12 manual weights and our method on three different datasets with varying noise rates. The green and red fonts represent the lowest and highest retrieval performance of the 12 manually weighted combinations of α NCE + β RCE, respectively.

Table 1. mAPs (%) (mean ± standard deviation) on the UCMD dataset for adaptive weights versus manual weights.

Loss	**The Noise Rate of UCMD**				
	Clean (0%)	**5.0%**	**10.0%**	**20.0%**	**30%**
CE	97.09 ± 0.37	94.37 ± 0.02	89.81 ± 0.40	78.89 ± 0.87	74.58 ± 2.99
0.1 NCE + 0.1 RCE	95.82 ± 0.24	95.49 ± 0.85	94.16 ± 0.64	90.81 ± 0.66	84.91 ± 1.45
0.1 NCE + 1 RCE	96.63 ± 0.24	95.12 ± 1.02	93.59 ± 1.00	91.01 ± 0.66	85.79 ± 0.91
0.1 NCE + 10 RCE	96.64 ± 0.48	95.01 ± 0.74	93.54 ± 0.15	92.33 ± 0.67	87.48 ± 0.52
0.1 NCE + 100 RCE	96.66 ± 0.39	94.01 ± 0.73 [2]	94.13 ± 0.96	89.91 ± 0.67 [2]	84.14 ± 1.04 [2]
1 NCE + 0.1 RCE	95.44 ± 0.50 [2]	95.46 ± 0.66	94.61 ± 0.60	92.56 ± 0.31 [1]	88.25 ± 0.74
1 NCE + 1 RCE	95.95 ± 0.37	95.63 ± 0.17	94.18 ± 0.39	90.12 ± 0.50	86.17 ± 0.90
1 NCE + 10 RCE	96.64 ± 0.49	95.58 ± 0.39	94.50 ± 0.84	91.67 ± 0.93	86.39 ± 0.59
1 NCE + 100 RCE	96.80 ± 0.45	95.33 ± 0.98	94.56 ± 0.45	91.74 ± 0.29	87.90 ± 0.68
10 NCE + 0.1 RCE	96.77 ± 0.70	95.74 ± 0.29 [1]	93.28 ± 0.71 [2]	91.10 ± 1.67	89.19 ± 0.67 [1]
10 NCE + 1 RCE	96.82 ± 0.59 [1]	95.65 ± 0.42	94.79 ± 0.71 [1]	92.43 ± 1.26	89.15 ± 0.68
10 NCE + 10 RCE	96.54 ± 0.57	95.73 ± 0.93	94.63 ± 0.79	91.23 ± 1.19	87.42 ± 1.69
10 NCE + 100 RCE	96.47 ± 0.57	94.78 ± 0.61	94.33 ± 0.72	91.28 ± 1.45	86.44 ± 0.99
A-NCE + RCE (ours)	97.00 ± 0.87 *	96.51 ± 0.40 *	95.02 ± 0.51 *	93.00 ± 0.32 *	90.01 ± 0.88 *

* Represents the highest retrieval precision at the same noise rate, and superscripts [1] and [2] represent the highest and lowest retrieval performance of the 12 manually weighted combinations of α NCE + β RCE, respectively.

Table 2. mAPs (%) (mean ± standard deviation) on the AID dataset for adaptive weights versus manual weights.

Loss	The Noise Rate of AID				
	Clean (0%)	**5.0%**	**10.0%**	**20.0%**	**30%**
CE	93.17 ± 0.94	90.07 ± 0.39	84.15 ± 0.46	71.95 ± 0.15	69.45 ± 3.41
0.1 NCE + 0.1 RCE	92.58 ± 0.62	92.06 ± 0.40	90.90 ± 0.48	88.38 ± 0.18	85.99 ± 0.61
0.1 NCE + 1 RCE	92.89 ± 0.48	92.13 ± 0.27	91.30 ± 0.22 ^{1}	88.06 ± 0.47	85.71 ± 0.32
0.1 NCE + 10 RCE	92.34 ± 0.44	91.69 ± 0.22	90.82 ± 0.28	88.63 ± 0.30	84.79 ± 0.82
0.1 NCE + 100 RCE	92.93 ± 0.42	91.91 ± 0.21	89.80 ± 0.46 ^{2}	88.34 ± 0.63	84.28 ± 0.20 ^{2}
1 NCE + 0.1 RCE	92.55 ± 0.27	92.43 ± 0.11 ^{1}	91.01 ± 0.44	89.74 ± 0.43	86.94 ± 0.37 ^{1}
1 NCE + 1 RCE	92.39 ± 0.32	91.98 ± 0.48	91.05 ± 0.53	89.37 ± 0.36	86.53 ± 0.11
1 NCE + 10 RCE	91.95 ± 0.30 ^{2}	92.36 ± 0.32	91.25 ± 0.33	89.06 ± 0.75	85.50 ± 0.32
1 NCE + 100 RCE	92.60 ± 0.26	92.38 ± 0.57	90.93 ± 0.54	87.85 ± 0.88 ^{2}	85.51 ± 0.63
10 NCE + 0.1 RCE	93.25 ± 0.17 ^{1}	92.41 ± 0.75	90.26 ± 0.30	89.30 ± 0.99	86.78 ± 1.57
10 NCE + 1 RCE	92.76 ± 0.31	92.27 ± 0.12	90.89 ± 0.13	89.79 ± 0.68 ^{1}	86.53 ± 0.35
10 NCE + 10 RCE	92.94 ± 0.32	92.40 ± 0.11	91.08 ± 0.12	89.66 ± 0.30	86.63 ± 0.30
10 NCE + 100 RCE	92.89 ± 0.33	91.58 ± 0.13 ^{2}	91.15 ± 0.35	88.48 ± 0.40	85.14 ± 0.98
A-NCE + RCE (ours)	93.68 ± 0.38 *	92.74 ± 0.05 *	92.37 ± 0.22 *	90.22 ± 0.70 *	87.17 ± 0.54 *

* Represents the highest retrieval precision at the same noise rate, and superscripts ^{1} and ^{2} represent the highest and lowest retrieval performance of the 12 manually weighted combinations of α NCE + β RCE, respectively.

It can be seen from Tables 1–3 that the original APL method outperforms the traditional CE method in the presence of noisy labels. However, the weighting hyperparameters required for optimal performance vary across the three datasets and noise rates. This means that, as the dataset and noise rate change, the original APL method must be repeatedly tuned. This is a time-consuming process with uncertain results. Furthermore, it can also be seen that our method achieves the best performance in the three datasets at different noise rates. This indicates that our adaptive weighted method is effective for the robust APL method.

Table 3. mAPs (%) (mean ± standard deviation) on the NWPU dataset for adaptive weights versus manual weights.

Loss	The Noise Rate of NWPU				
	Clean (0%)	**5.0%**	**10.0%**	**20.0%**	**30%**
CE	90.99 ± 0.75	85.96 ± 0.82	81.57 ± 1.84	80.41 ± 0.52	77.07 ± 0.03
0.1 NCE + 0.1 RCE	89.55 ± 0.42	89.47 ± 0.53	87.92 ± 0.63	86.41 ± 0.26	84.15 ± 0.26
0.1 NCE + 1 RCE	89.33 ± 0.28	89.47 ± 0.12	88.04 ± 0.54	86.99 ± 0.42	84.16 ± 0.59
0.1 NCE + 10 RCE	89.58 ± 0.36	88.68 ± 0.38	87.86 ± 0.29 ^{2}	86.73 ± 0.57	84.33 ± 0.31
0.1 NCE + 100 RCE	88.92 ± 0.61	88.67 ± 0.73	88.34 ± 0.28	85.80 ± 0.16 ^{2}	84.90 ± 0.62
1 NCE + 0.1 RCE	90.13 ± 0.27	88.96 ± 0.75	88.91 ± 0.41	87.32 ± 0.42	85.27 ± 0.36
1 NCE + 1 RCE	89.89 ± 0.14	89.04 ± 0.47	88.32 ± 0.26	87.11 ± 0.11	83.82 ± 0.45 ^{2}
1 NCE + 10 RCE	89.46 ± 0.61	88.84 ± 0.40	88.53 ± 0.15	85.93 ± 0.09	84.18 ± 0.52
1 NCE + 100 RCE	88.89 ± 0.12 ^{2}	88.74 ± 0.39	88.02 ± 0.55	86.38 ± 0.33	85.02 ± 0.40
10 NCE + 0.1 RCE	90.17 ± 0.83 ^{1}	89.68 ± 0.19 ^{1}	89.62 ± 0.35 ^{1}	87.70 ± 0.11 ^{1}	85.51 ± 0.78 ^{1}
10 NCE + 1 RCE	89.62 ± 0.31	88.69 ± 0.19	88.60 ± 0.64	87.01 ± 0.50	84.69 ± 0.58
10 NCE + 10 RCE	89.17 ± 0.25	88.60 ± 0.44 ^{2}	88.11 ± 0.10	86.38 ± 0.35	84.74 ± 0.36
10 NCE + 100 RCE	89.76 ± 0.10	88.96 ± 0.24	88.19 ± 0.87	86.79 ± 0.34	84.26 ± 0.40
A-NCE + RCE (ours)	90.60 ± 0.52 *	90.12 ± 0.53 *	90.04 ± 0.55 *	88.37 ± 0.29 *	86.88 ± 0.45 *

* Represents the highest retrieval precision at the same noise rate, and superscripts ^{1} and ^{2} represent the highest and lowest retrieval performance of the 12 manually weighted combinations of α NCE + β RCE, respectively.

In Table 4, we compare the image classification accuracy between adaptive weights and manual weights. It is evident that our method is more conducive to improving the feature extraction ability of the model and achieves higher image classification accuracy compared to manual weights. Therefore, our method is also suitable for image-level computer vision tasks based on high-level features such as image classification.

Table 4. Comparison of image classification accuracy between manual and automatic weighting methods on the UCMD dataset with 20% noise.

Loss	Noise Rates				
	Clean (0%)	**5.0%**	**10.0%**	**20.0%**	**30%**
NCE + RCE	98.05 ± 0.16	97.58 ± 0.44	96.58 ± 0.24	95.92 ± 0.52	95.27 ± 0.31
A-NCE + RCE (ours)	98.57 ± 0.40	98.70 ± 0.49 *	97.72 ± 0.09 *	96.97 ± 0.68 *	96.68 ± 0.42 *

* Represents the highest retrieval precision.

4.3. Comparison with Various SOTA Losses

In these experiments, we apply our method to four types of APLs on the UCMD dataset with 20% noise rate (results in Table 5) and compare them with seven state-of-the-art robust losses on the NWPU dataset with 20% noise rate (results in Table 6). These methods are described below.

(1) MAE [13]: It is a passive and symmetric loss, as described in Section 1. Although it can maintain gradient stability for different input values, its training is limited due to slow convergence.

(2) GCE [14]: It is also a passive and symmetric loss, as described in Section 1. Its robustness is achieved by combining the MAE with the CE, and it is only robust when reduced to the MAE loss.

(3) RCE [15]: It can be seen as the reverse version of the CE, as it exchanges the positions of the predicted probability and the one-hot coding in the formula of the cross-entropy loss function. However, it also converges slowly.

(4) SCE [15]: It combines the CE loss with the RCE. Its robustness and convergence stability are guaranteed by RCE and CE, respectively. However, it requires the adjustment of two hyperparameters.

(5) ACE [16]: It uses the predicted probabilities p_t of the true labels of the samples to adaptively de-termine the two weights in SCE. As the p_t of the sample tends toward zero, it gradually transforms into RCE.

(6) AUL [31]: It is a noise robust function that is an asymmetric version of the unhinged loss [27].

(7) AEL [31]: It is an asymmetric noise-robust function, which assumes that the noise distribution in the data satisfies the clean label domination assumption.

Table 5. mAPs (%) (mean ± standard deviation) on the UCMD dataset with 20% noise using 4 types of APLs.

The Weight of APL	The Type of APL			
	NCE + RCE	**NCE + MAE**	**NFL + RCE**	**NFL + MAE**
$\alpha = 0.1,\ \beta = 0.1$	90.81 ± 0.66	90.75 ± 1.36	91.10 ± 0.62	90.05 ± 0.18 [2]
$\alpha = 0.1,\ \beta = 1$	91.01 ± 0.66	91.67 ± 0.79	91.86 ± 1.07	92.07 ± 0.54 [1]
$\alpha = 0.1,\ \beta = 10$	92.33 ± 0.67	90.70 ± 0.42	91.52 ± 1.10	90.87 ± 0.59
$\alpha = 0.1,\ \beta = 100$	89.91 ± 0.67 [2]	91.59 ± 1.48	91.24 ± 0.13	90.77 ± 0.39
$\alpha = 1,\ \beta = 0.1$	92.56 ± 0.31 [1]	91.34 ± 2.49	91.39 ± 0.13	91.15 ± 1.94
$\alpha = 1,\ \beta = 1$	90.12 ± 0.50	91.29 ± 0.23	90.57 ± 0.51	91.61 ± 0.68
$\alpha = 1,\ \beta = 10$	91.67 ± 0.93	91.76 ± 1.31 [1]	90.00 ± 0.71 [2]	91.33 ± 1.08
$\alpha = 1,\ \beta = 100$	91.74 ± 0.29	90.48 ± 0.95	91.43 ± 0.98	90.76 ± 0.64
$\alpha = 10,\ \beta = 0.1$	91.10 ± 1.67	88.73 ± 1.84 [2]	90.88 ± 2.83	90.23 ± 0.26
$\alpha = 10,\ \beta = 1$	92.43 ± 1.26	91.68 ± 0.51	91.88 ± 0.41 [1]	91.95 ± 0.62
$\alpha = 10,\ \beta = 10$	91.23 ± 1.19	91.64 ± 1.04	91.82 ± 0.74	90.42 ± 1.41
$\alpha = 10,\ \beta = 100$	91.28 ± 1.45	90.51 ± 1.54	90.94 ± 1.37	91.12 ± 0.85
A-APL (ours)	93.00 ± 0.32 *	92.28 ± 0.88 *	92.03 ± 0.79 *	92.54 ± 0.29 *

* Represents the highest retrieval precision, and superscripts [1] and [2] represent the highest and lowest retrieval performance of the 12 manually weighted combinations of α NCE + β RCE, respectively.

Table 6. mAPs (%) (mean ± standard deviation) of comparison with robust losses on the NWPU dataset with 20% noise.

Methods	mAP	Methods	mAP
CE	80.41 ± 0.52	NCE + RCE [17]	87.70 ± 0.11
MAE [13]	86.53 ± 0.13	NCE + MAE [17]	85.82 ± 0.53
GCE [14]	87.97 ± 0.48	NFL + RCE [17]	88.04 ± 0.32
RCE [15]	86.39 ± 0.37	NFL + MAE [17]	84.07 ± 1.20
SCE [15]	86.51 ± 0.19	A-NCE + RCE (ours)	88.37 ± 0.29 *
ACE [16]	86.52 ± 0.33	A-NCE + MAE (ours)	88.95 ± 0.45 *
AUL [31]	86.74 ± 0.82	A-NFL + RCE (ours)	88.76 ± 0.25 *
AEL [31]	85.79 ± 0.08	A-NFL + MAE (ours)	88.49 ± 0.26 *

* Represents the highest retrieval precision.

As shown in Table 5, the four APL combinations also have different weighting parameters to achieve optimality, and our method still outperforms their 12 manually weighted combinations. Additionally, our method outperforms the best manually weighted methods by 0.67–4.42% and other robust loss methods by 0.4–2.56%.

It can be seen from Table 6 that our improved APL methods only yield a slight improvement by 0.4–0.98% over the GCE method. However, it is important to note that the GCE method requires manual determination of a hyperparameter in the (0, 1] interval, and unfortunately, the literature [14] does not provide a method for parameter selection. Compared to the ACE method, which also adaptively determines the hyperparameter, our improved APL methods show a more significant improvement of 1.85–2.43%. Moreover, our method is also superior to asymmetric loss functions such as AUL and AEL. The above results indicate that our weighted learning method is suitable for all types of APL methods and yields superior retrieval outcomes compared to other robust methods.

4.4. Efficiency and Backbone Analysis

The training time and floating point of operations (FLOPs) are used to evaluate the efficiency on the UCMD dataset with a 20% noise rate using different backbones. The experimental results are presented in Table 7. On the one hand, it is evident that the FLOPs of our improved APL method do not increase significantly. This is because AWNet is a shallow neural network with significantly fewer parameters than deep learning models. As a result, the training time of our improved APL method does not increase significantly compared to the original APL. Considering that the original APL method has to try at least 12 different weighting parameters, it actually takes longer to train than our improved APL. On the other hand, it can be seen that our improved APL method achieves better retrieval results than the original APL method under different deep learning models. This indicates that our method has better generalisation.

4.5. Ablation Experiment of Two Metrics

To verify the validity of the two metrics (RES and δ) of the AWNet, we perform an ablation experiment on the UCMD dataset with noise rates of 20% and 30%, respectively. In the ablation experiments, the retrieval effects of the metrics RES and δ alone as inputs of the AWNet are investigated. In addition, the retrieval effects of the three parameters prediction probability, entropy and standard deviation S, which make up RES and δ, are also examined alone as the input of the AWNet. Specifically, the prediction probability denotes the predicted probability of the model classifier (i.e., the output of the softmax output layer).

Table 8 displays the retrieval performance of the above experiments. The results indicate that using both RES and δ as inputs to AWNet can help the AWNet to better predict α and β, resulting in the highest retrieval accuracy. This finding confirms the validity of the two metrics.

Table 7. mAPs (%) (mean ± standard deviation) of different backbones on the UCMD dataset with 20% noise.

Backbone	Loss	mAP	Training Time (min)	FLOPs (G)
ResNet18	CE	69.48 ± 1.54	1.05	2.375
	NCE + RCE	87.21 ± 1.15	1.07 × 12	2.375
	A-NCE + RCE	87.78 ± 0.78 *	1.40	2.375
ResNet50	CE	78.89 ± 0.87	1.83	5.368
	NCE + RCE	92.56 ± 0.31	1.88 × 12	5.368
	A-NCE + RCE	93.00 ± 0.32 *	2.98	5.368
ResNet101	CE	79.55 ± 0.79	2.73	10.230
	NCE + RCE	89.72 ± 1.34	3.33 × 12	10.230
	A-NCE + RCE	91.94 ± 0.67 *	4.75	10.230
DenseNet169	CE	87.46 ± 0.17	2.40	4.436
	NCE + RCE	94.93 ± 0.55	2.40 × 12	4.436
	A-NCE + RCE	95.45 ± 0.50 *	4.1	4.436
MobileNetV3_large	CE	81.18 ± 0.86	0.95	0.292
	NCE + RCE	89.63 ± 0.86	0.97 × 12	0.292
	A-NCE + RCE	90.99 ± 0.33 *	1.37	0.292
MobileNetV3_small	CE	75.43 ± 1.13	0.68	0.076
	NCE + RCE	84.59 ± 0.27	0.68 × 12	0.076
	A-NCE + RCE	87.80 ± 0.15 *	0.88	0.076

* Represents the highest retrieval precision.

Table 8. mAPs (%) (mean ± standard deviation) of different inputs of AWNet on the UCMD dataset with 20% and 30% noise.

Metrics	20%	30%
Prediction probability	91.17 ± 0.58	88.52 ± 0.05
Entropy	91.83 ± 0.37	86.82 ± 1.95
S	91.39 ± 0.74	87.38 ± 1.50
δ	91.14 ± 0.59	87.94 ± 0.78
RES	92.44 ± 0.36	86.47 ± 1.42
RES + δ	93.00 ± 0.32 *	90.01 ± 0.88 *

* Represents the highest retrieval precision.

Additionally, to assess the rationality of the pretraining epochs t_p in this paper, we conduct ablation experiments on the UCMD dataset with 20% noise. The pretraining epochs t_p are set to 1, 2, 3, 4 and 5, respectively. The experimental results are shown in Table 9. The results show that the retrieval performance achieved the best when $t_p = 3$. Too few pretraining epochs (e.g., 1) may lead to underfitting of the model and affect the reliability of RES and δ. Therefore, in other experiments, the pretraining epoch is set to 3.

Table 9. mAPs (%) (mean ± standard deviation) of different pretraining epochs t_p on the UCMD dataset with 20% noise.

t_p = 1	t_p = 2	t_p = 3	t_p = 4	t_p = 5
91.48 ± 1.05	92.25 ± 0.77	93.00 ± 0.32 *	92.88 ± 0.47	92.55 ± 0.68

* Represents the highest retrieval precision.

To determine the optimal structure for the AWNet, we evaluate the model performance with varying numbers of hidden layers and neurons per layer on the UCMD dataset with 20% noise. Specifically, we test the model's performance with one, two and three hidden layers and with 50 and 100 neurons per layer, respectively. The experimental results are presented in Table 10. The results indicate that using one hidden layer yields a better retrieval accuracy compared to using two or three hidden layers while also requiring fewer parameters and thus reducing calculation costs. In addition, a hidden layer with 100 neurons is better suited for fitting the relationship between training samples and weighting hyperparameters, resulting in a higher retrieval accuracy.

Table 10. mAPs (%) (mean ± standard deviation) of different numbers of hidden layers on the UCMD dataset with 20% noise.

	The Number of Hidden Layers		
The Number of Neurons in Each Hidden Layer	**1**	**2**	**3**
50	90.65 ± 0.72	91.99 ± 0.75	91.37 ± 0.97
100	93.00 ± 0.32 *	91.41 ± 0.27	92.06 ± 0.18

* Represents the highest retrieval precision.

5. Conclusions

In this paper, we propose an adaptively weighted method based on active passive loss (APL) for remote sensing image retrieval. To automatically determine the weight hyperparameters of active losses and passive losses of different samples, we first design or select two metrics to measure the sample complexity and noise level based on entropy, standard deviation and predicted probability difference. Then, an adaptive weighted learning network (AWNet) based on the multilayer perceptron is designed to automatically predict the weighting parameters.

In order to verify the effectiveness and portability of our method, four groups of experiments are designed. First of all, the experimental results show that the retrieval accuracy of our method is better than that of 12 manual weighting combinations on three datasets: UCMD, AID and NWPU with five noise rates. Secondly, compared with seven other state-of-the-art robust losses, our method achieves the best performance, and the mAPs are improved by 0.4% to 2.56%. In the third group of experiments, we compare the retrieval accuracy and model complexity of our method using six different backbones. The results show that our method has excellent performance and good portability without increasing the computational cost too much. Finally, we proved the rationality of the AWNet's input metrics and structure through several groups of ablation experiments. In addition, the results show that our method achieves better image classification accuracy than manual weighting. Therefore, our process of adaptively learning the weighting parameters can benefit other areas such as image classification and segmentation with noisy labels.

Although the existing retrieval models have achieved excellent retrieval performance on single-domain datasets, they are difficult to generalise to test datasets in other domains. To address this issue, many scholars have proposed remote sensing image retrieval methods that enhance the generalisation performance of models on different data sources. For example, Wang et al. [4] implemented unsupervised cross-domain remote sensing image retrieval by using pseudo-label self-training and consistency regulation. However, those domain adaptation methods lack research on noise labels. Therefore, our future work will be concentrated on the effect of noise samples on domain adaptation remote sensing image retrieval.

Author Contributions: Conceptualisation, X.T., D.H., S.W. and H.X.; methodology, X.T., D.H. and X.L.; validation, X.T.; writing—original draft preparation, X.T., D.H. and H.X.; funding acquisition, D.H. and H.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded in part by the National Natural Science Foundation of China under grant 42171457, in part by the Shandong Province Excellent Youth Fund under grant ZR2022YQ36 and in part by the Yunnan Provincial Philosophy and Social Science Innovation Team Construction Project under grant 2023CX02.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Acknowledgments: The authors would like to thank Shawn Newsam (https://vision.ucmerced.edu/datasets/ (accessed on 25 January 2024)), Wuhan University and Huazhong University of Science and Technology (https://captain-whu.github.io/AID/ (accessed on 25 January 2024)) and Northwestern Polytechnical University (http://pan.baidu.com/s/1mifR6tU (accessed on 25 January 2024)) for making the remote sensing image datasets publicly available. The authors would also like to thank the anonymous reviewers for their comments to improve this paper.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Li, D.-R.; Shao, Z.-F.; Zhang, R.-Q. Advances of geo-spatial intelligence at LIESMARS. *Geo-Spat. Inf. Sci.* **2020**, *23*, 40–51. [CrossRef]
2. Ye, F.-M.; Xiao, H.; Zhao, X.-Q.; Dong, M.; Luo, W.; Min, W.-D. Remote sensing image retrieval using convolutional neural network features and weighted distance. *IEEE Geosci. Remote Sens. Lett.* **2018**, *15*, 1535–1539. [CrossRef]
3. Dubey, S.R. A Decade Survey of Content Based Image Retrieval Using Deep Learning. *IEEE Trans. Circuits Syst. Video Technol.* **2022**, *32*, 2687–2704. [CrossRef]
4. Hou, D.; Wang, S.; Tian, X.; Xing, H. PCLUDA: A Pseudo-Label Consistency Learning-Based Unsupervised Domain Adaptation Method for Cross-Domain Optical Remote Sensing Image Retrieval. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5600314. [CrossRef]
5. Liu, E.; Zhang, X.; Xu, X.; Fan, S. Slice-feature based deep hashing algorithm for remote sensing image retrieval. *Infrared Phys. Technol.* **2020**, *107*, 103299. [CrossRef]
6. Hou, D.-Y.; Miao, Z.-L.; Xing, H.-Q.; Wu, H. V-RSIR: An Open Access Web-Based Image Annotation Tool for Remote Sensing Image Retrieval. *IEEE Access* **2019**, *7*, 83852–83862. [CrossRef]
7. Hou, D.-Y.; Miao, Z.-L.; Xing, H.-Q.; Wu, H. Two novel benchmark datasets from ArcGIS and bing world imagery for remote sensing image retrieval. *Int. J. Remote Sens.* **2021**, *42*, 220–238. [CrossRef]
8. Jin, P.; Xia, G.-S.; Hu, F.; Lu, Q.-K.; Zhang, L.-P. AID++: An updated version of AID on scene classification. In Proceedings of the 2018 IEEE IGARSS, 2018 IEEE International Geoscience and Remote Sensing Symposium, Valencia, Spain, 22–27 July 2018; pp. 4721–4724.
9. Song, H.; Kim, M.; Park, D.; Shin, Y.; Lee, J.-G. Learning from Noisy Labels with Deep Neural Networks: A Survey. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *34*, 8135–8153. [CrossRef]
10. Miao, Q.; Wu, X.; Xu, C.; Zuo, W.; Meng, Z. On better detecting and leveraging noisy samples for learning with severe label noise. *Pattern Recognit.* **2023**, *136*, 109210. [CrossRef]
11. Li, J.; Wong, Y.; Zhao, Q.; Kankanhalli, M.S. Learning to Learn from Noisy Labeled Data. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 5046–5054.
12. Kang, J.; Fernandez-Beltran, R.; Kang, X.; Ni, J.; Plaza, A. Noise-Tolerant Deep Neighborhood Embedding for Remotely Sensed Images with Label Noise. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 2551–2562. [CrossRef]
13. Ghosh, A.; Kumar, H.; Sastry, P.S. Robust loss functions under label noise for deep neural networks. In Proceedings of the AAAI Conference on Artificial Intelligence, San Francisco, CA, USA, 4–9 February 2017.
14. Zhang, Z.; Sabuncu, M. Generalized cross entropy loss for training deep neural networks with noisy labels. In Proceedings of the NeurIPS, Montréal, QC, Canada, 4–5 December 2018.
15. Wang, Y.; Ma, X.; Chen, Z.; Luo, Y.; Yi, J.; Bailey, J. Symmetric Cross Entropy for Robust Learning with Noisy Labels. In Proceedings of the 2019 IEEE ICCV, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 322–330.
16. Chen, H.D.; Tan, W.M.; Li, J.C.; Guan, P.F.; Wu, L.J.; Yan, B.; Li, J.; Wang, Y.F. Adaptive Cross Entropy for ultrasmall object detection in Computed Tomography with noisy labels. *Comput. Biol. Med.* **2022**, *147*, 105763. [CrossRef]
17. Ma, X.-J.; Huang, H.-X.; Wang, Y.-S.; Romano, S.; Erfani, S.; Bailey, J. Normalized Loss Functions for Deep Learning with Noisy Labels. In Proceedings of the ICML, PMLR, Vienna, Austria, 12–18 July 2020; pp. 6543–6553.
18. Zhou, W.X.; Newsam, S.; Li, C.M.; Shao, Z.F. PatternNet: A benchmark dataset for performance evaluation of remote sensing image retrieval. *ISPRS J. Photogramm. Remote Sens.* **2018**, *145*, 197–209. [CrossRef]
19. Wang, S.; Hou, D.; Xing, H. A Self-Supervised-Driven Open-Set Unsupervised Domain Adaptation Method for Optical Remote Sensing Image Scene Classification and Retrieval. *IEEE Trans. Geosci. Remote Sens.* **2023**, *61*, 5605515. [CrossRef]
20. Hou, D.; Wang, S.; Tian, X.; Xing, H. An Attention-Enhanced End-to-End Discriminative Network with Multiscale Feature Learning for Remote Sensing Image Retrieval. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2022**, *15*, 8245–8255. [CrossRef]
21. Wang, S.Y.; Hou, D.Y.; Xing, H.Q. A novel multi-attention fusion network with dilated convolution and label smoothing for remote sensing image retrieval. *Int. J. Remote Sens.* **2022**, *43*, 1306–1322. [CrossRef]
22. Li, Y.; Zhang, Y.; Zhu, Z. Error-tolerant deep learning for remote sensing image scene classification. *IEEE Trans. Cybern.* **2020**, *51*, 1756–1768. [CrossRef] [PubMed]

23. Damodaran, B.B.; Flamary, R.; Seguy, V.; Courty, N. An entropic optimal transport loss for learning deep neural networks under label noise in remote sensing images. *Comput. Vis. Image Underst.* **2020**, *191*, 102863. [CrossRef]
24. Manwani, N.; Sastry, P.S. Noise Tolerance Under Risk Minimization. *IEEE Trans. Cybern.* **2013**, *43*, 1146–1151. [CrossRef]
25. Jiang, J.; Ma, J.; Wang, Z.; Chen, C.; Liu, X. Hyperspectral Image Classification in the Presence of Noisy Labels. *IEEE Trans. Geosci. Remote Sens.* **2019**, *57*, 851–865. [CrossRef]
26. Lee, K.H.; He, X.; Zhang, L.; Yang, L. CleanNet: Transfer Learning for Scalable Image Classifier Training with Label Noise. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 5447–5456.
27. Van Rooyen, B.; Menon, A.; Williamson, R.C. Learning with symmetric label noise: The importance of being unhinged. In Proceedings of the Advances in Neural Information Processing Systems 28 (NIPS 2015), Montreal, QC, Canada 7–12 December 2015; Volume 28.
28. Charoenphakdee, N.; Lee, J.; Sugiyama, M. On symmetric losses for learning from corrupted labels. In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019; pp. 961–970.
29. Zhou, X.; Liu, X.; Wang, C.; Zhai, D.; Jiang, J.; Ji, X. Learning with noisy labels via sparse regularization. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Montreal, QC, Canada, 11–17 October 2021; pp. 72–81.
30. Kim, Y.; Yim, J.; Yun, J.; Kim, J. NLNL: Negative Learning for Noisy Labels. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019; pp. 101–110.
31. Zhou, X.; Liu, X.; Zhai, D.; Jiang, J.; Ji, X. Asymmetric Loss Functions for Noise-Tolerant Learning: Theory and Applications. *IEEE Trans. Pattern Anal. Mach. Intell.* **2023**, *45*, 8094–8109. [CrossRef]
32. Mittal, P.; Sharma, A.; Singh, R. Deformable patch-based-multi-layer perceptron Mixer model for forest fire aerial image classification. *J. Appl. Remote Sens.* **2023**, *17*, 022203. [CrossRef]
33. Akbari, D.; Ashrafi, A.; Attarzadeh, R. A New Method for Object-Based Hyperspectral Image Classification. *J. Indian Soc. Remote Sens.* **2022**, *50*, 1761–1771. [CrossRef]
34. Gong, N.; Zhang, C.; Zhou, H.; Zhang, K.; Wu, Z.; Zhang, X. Classification of hyperspectral images via improved cycle-MLP. *IET Comput. Vis.* **2022**, *16*, 468–478. [CrossRef]
35. Huang, K.; Tian, C.; Li, G. Bidirectional mutual guidance transformer for salient object detection in optical remote sensing images. *Int. J. Remote Sens.* **2023**, *44*, 4016–4033. [CrossRef]
36. Wang, L.; Li, H. HMCNet: Hybrid Efficient Remote Sensing Images Change Detection Network Based on Cross-Axis Attention MLP and CNN. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 5236514. [CrossRef]
37. Shi, G.; Mei, Y.; Wang, X.; Yang, Q. DAHT-Net: Deformable Attention-Guided Hierarchical Transformer Network Based on Remote Sensing Image Change Detection. *IEEE Access* **2023**, *11*, 103033–103043. [CrossRef]
38. Tolstikhin, I.O.; Houlsby, N.; Kolesnikov, A.; Beyer, L.; Zhai, X.; Unterthiner, T.; Yung, J.; Steiner, A.; Keysers, D.; Uszkoreit, J. Mlp-mixer: An all-mlp architecture for vision. *Adv. Neural Inf. Process. Syst.* **2021**, *34*, 24261–24272.
39. Ding, X.; Xia, C.; Zhang, X.; Chu, X.; Han, J.; Ding, G. Repmlp: Re-parameterizing convolutions into fully-connected layers for image recognition. *arXiv* **2021**, arXiv:2105.01883.
40. Zhang, C.; Pan, X.; Li, H.; Gardiner, A.; Sargent, I.; Hare, J.; Atkinson, P.M. A hybrid MLP-CNN classifier for very fine resolution remotely sensed image classification. *ISPRS J. Photogramm. Remote Sens.* **2018**, *140*, 133–144. [CrossRef]
41. Hospedales, T.; Antoniou, A.; Micaelli, P.; Storkey, A. Meta-Learning in Neural Networks: A Survey. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 5149–5169. [CrossRef]
42. Zhao, Q.-H.; Hu, W.; Huang, Y.-Y.; Zhang, F. P-DIFF plus: Improving learning classifier with noisy labels by Noisy Negative Learning loss. *Neural Netw.* **2021**, *144*, 1–10. [CrossRef] [PubMed]
43. Yang, Y.; Newsam, S. Bag-of-visual-words and spatial extensions for land-use classification. In Proceedings of the 18th SIGSPATIAL, International Conference on Advances in Geographic Information Systems, San Jose, CA, USA, 2–5 November 2010; pp. 270–279.
44. Xia, G.-S.; Hu, J.-W.; Hu, F.; Shi, B.-G.; Bai, X.; Zhong, Y.-F.; Zhang, L.-P.; Lu, X.-Q. AID: A Benchmark Data Set for Performance Evaluation of Aerial Scene Classification. *IEEE Trans. Geosci. Remote Sens.* **2017**, *55*, 3965–3981. [CrossRef]
45. Cheng, G.; Han, J.-W.; Lu, X.-Q. Remote Sensing Image Scene Classification: Benchmark and State of the Art. *Proc. IEEE* **2017**, *105*, 1865–1883. [CrossRef]
46. He, K.-M.; Zhang, X.-Y.; Ren, S.-Q.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
47. Huang, G.; Liu, Z.; Maaten, L.V.D.; Weinberger, K.Q. Densely Connected Convolutional Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2017, CVPR, Honolulu, HI, USA, 21–26 July 2017; pp. 2261–2269.
48. Howard, A.; Sandler, M.; Chen, B.; Wang, W.; Chen, L.C.; Tan, M.; Chu, G.; Vasudevan, V.; Zhu, Y.; Pang, R.; et al. Searching for MobileNetV3. In Proceedings of the 2019 IEEE ICCV, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 1314–1324.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.