diversity

MDPI

Article

Improved Wildlife Recognition through Fusing Camera Trap Images and Temporal Metadata

Lei Liu [1,2], Chao Mou [1,2,3,*] and Fu Xu [1,2,3,*]

[1] School of Information Science and Technology, Beijing Forestry University, Beijing 100083, China; liulei815@bjfu.edu.cn
[2] Engineering Research Center for Forestry-Oriented Intelligent Information Processing of National Forestry and Grassland Administration, Beijing 100083, China
[3] State Key Laboratory of Efficient Production of Forest Resources, Beijing 100083, China
* Correspondence: chao_m@bjfu.edu.cn (C.M.); xufu@bjfu.edu.cn (F.X.)

Abstract: Camera traps play an important role in biodiversity monitoring. An increasing number of studies have been conducted to automatically recognize wildlife in camera trap images through deep learning. However, wildlife recognition by camera trap images alone is often limited by the size and quality of the dataset. To address the above issues, we propose the Temporal-SE-ResNet50 network, which aims to improve wildlife recognition accuracy by exploiting the temporal information attached to camera trap images. First, we constructed the SE-ResNet50 network to extract image features. Second, we obtained temporal metadata from camera trap images, and after cyclical encoding, we used a residual multilayer perceptron (MLP) network to obtain temporal features. Finally, the image features and temporal features were fused in wildlife identification by a dynamic MLP module. The experimental results on the Camdeboo dataset show that the accuracy of wildlife recognition after fusing the image and temporal information is about 93.10%, which is an improvement of 0.53%, 0.94%, 1.35%, 2.93%, and 5.98%, respectively, compared with the ResNet50, VGG19, ShuffleNetV2-2.0x, MobileNetV3-L, and ConvNeXt-B models. Furthermore, we demonstrate the effectiveness of the proposed method on different national park camera trap datasets. Our method provides a new idea for fusing animal domain knowledge to further improve the accuracy of wildlife recognition, which can better serve wildlife conservation and ecological research.

Keywords: wildlife recognition; ResNet50; temporal metadata; attention mechanism; data fusion

Citation: Liu, L.; Mou, C.; Xu, F. Improved Wildlife Recognition through Fusing Camera Trap Images and Temporal Metadata. *Diversity* **2024**, *16*, 139. https://doi.org/10.3390/d16030139

Academic Editor: Michael Wink

Received: 18 December 2023
Revised: 19 February 2024
Accepted: 20 February 2024
Published: 23 February 2024

1. Introduction

Long-term monitoring of wildlife plays an important role in biodiversity conservation research [1]. Camera traps have been widely used in wildlife monitoring due to their non-invasive nature and low cost [2]. Camera trap data are used in many studies [3,4] for animal behavior identification [3] and abundance estimation [4]. These studies are based on identifying the species pictured in camera trap footage from camera trap images. It is more helpful to start the above studies when the recognition is more accurate.

Wildlife identification research based on deep learning has acquired more and more attention due to the benefits of automatically extracting wildlife-related information and the effective processing of a large number of images [5]. Gomez Villa et al. [6] selected 26 animals from the Snapshot Serengeti dataset and evaluated the potential of deep convolutional neural network frameworks such as AlexNet, VGGNet, GoogLeNet, and ResNet for the species identification task. The recognition accuracy of top-1 is 35.4% when the training dataset is unbalanced and contains empty images and 88.9% when the dataset is balanced and the images contain only foreground animals. Zualkernan et al. [7] proposed an edge-side wildlife recognition architecture using the Internet of Things (IoT) and the Xception model to recognize wildlife images captured by camera traps and transmit the

recognition results in real-time to a remote mobile application. Furthermore, by comparing the accuracy of VGG16, ResNet50, and self-trained networks in recognizing animal species such as snakes, lizards, and toads in camera-captured images, it was demonstrated that both ResNet50 and VGG trained using transfer learning outperform the self-trained model [8].

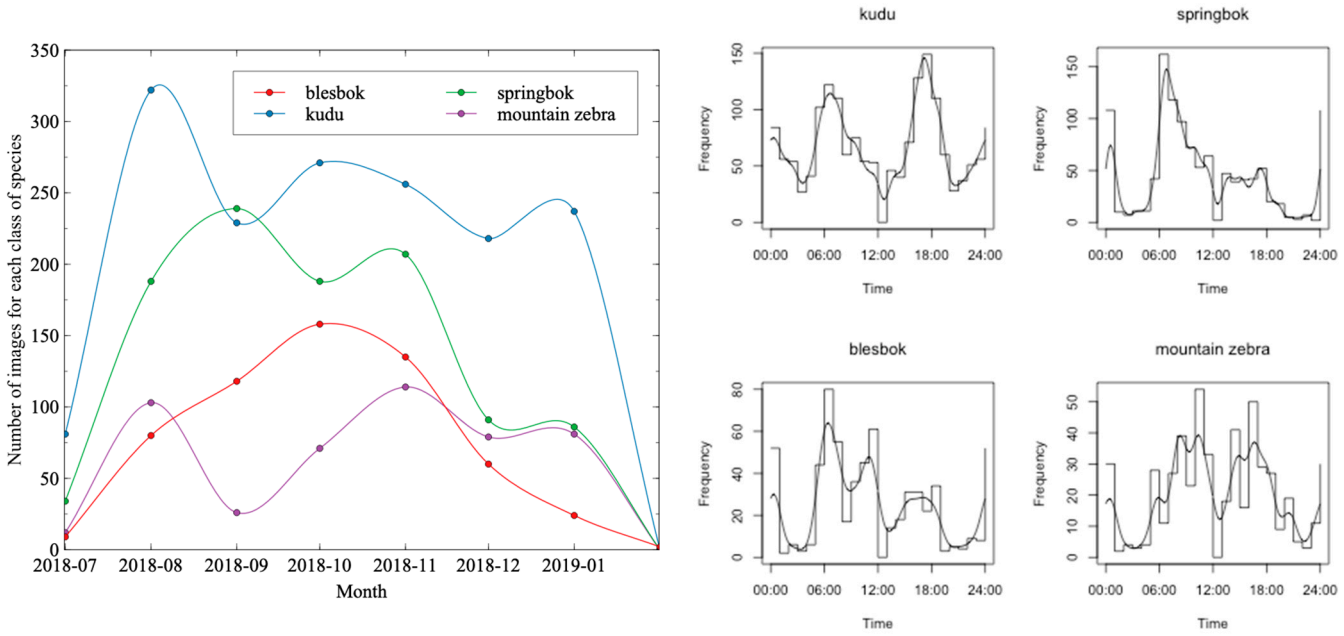
To enhance the performance of wildlife recognition, Xie et al. [9] proposed an integrated SE-ResNeXt model based on a multi-scale animal feature extraction module and a vision attention module, which enhances the feature extraction capability of the model and improved the recognition accuracy on a self-constructed wildlife dataset from 88.1% to 92.5%. Yang et al. [10] improved the accuracy of YOLOv5s from 72.6% to 89.4% by introducing the channel attention mechanism and the self-attention mechanism. Zhang et al. [11] designed a deep joint adaptation network for wildlife image recognition, which improved the generalization ability of the model in open scenarios and increased the recognition accuracy of 11 animal species from 54.6% to 58.2%.

In addition to improving model recognition performance by modifying the network structure and training strategy, some studies have also improved model recognition performance through data enhancement methods. Ahmed et al. [12] used camera-captured images with noisy labels, which turned some of the correct labels in the training set into wrong labels, to classify animals and improved the accuracy of recognition by selecting the largest prediction from multiple trained networks. Zhong et al. [13] proposed a data enhancement strategy that integrates image synthesis and regional background suppression to improve the performance of wildlife recognition and combines them with a model compression strategy to provide a lightweight recognition model that enables real-time monitoring on edge devices. Tan et al. [14] evaluated the YOLOv5, Cascade R-CNN, and FCOS models using daytime and nighttime camera trap data, demonstrating that models trained jointly by day and night can improve the accuracy of animal classification compared to that when using only nighttime data.

Currently, most wildlife recognition methods can only use camera trap image data for classification. However, limited by factors such as the shooting angle, animal pose, background environment, lighting conditions, etc., some animals are difficult to distinguish in camera trap images alone. In animal identification tasks using citizen science images, contextual information such as the climate, date, and location that accompanies the acquisition of citizen science imagery is used to identify the wildlife. Terry et al. [15] developed a multi-input neural network model that fuses contextual metadata and images to identify ladybird species in the British Isles, UK, demonstrating that deep learning models can effectively use contextual information to improve the top-1 accuracy of multi-input models from 48.2% to 57.3%. de Lutio et al. [16] utilized the spatial, temporal, and ecological contexts attached to most plant species' observation information to construct a digital taxonomist that improved accuracy from 73.48% to 79.12% compared to a model trained using only images. Mou et al. [17] used animals' visual features, for example, the color of a bird's feathers or the color of an animal's fur, to improve the recognition accuracy of a contrastive language–image pre-trained (CLIP) model on multiple animal datasets. Camera traps in national parks are capable of monitoring wildlife continuously for long periods of time with reduced human intervention and can provide complete information on animal rhythms. However, the use of animal rhythm information to aid wildlife recognition in camera trap images in national park scenes has not yet been explored.

Along with the massive camera trap image collection process, we obtain a lot of temporal metadata, including the date and time. These temporal metadata can reflect the activity rhythms of animals [15,18,19]. The quantity of data collected on various days throughout the year fluctuates due to variations in animal activity levels. Specifically, as shown in Figure 1, the number of camera trap images capturing kudus is greatest in August, whereas the highest number of camera trap images featuring blesboks is observed in October. Furthermore, the amount of data acquired during daylight hours varies due to distinct circadian rhythms. Based on the animal movement patterns fitted to the images

captured by the camera traps, it was found that springboks were most active from 06:00 h to 07:00 h, while kudus were most active from 17:00 h to 18:00 h.



(**a**) (**b**)

Figure 1. The kudu, springbok, blesbok, and mountain zebra are species belonging to the Camdeboo dataset. (**a**) Distribution of camera trap images in different months. (**b**) Activity patterns of different species at different times of the day fitted with Rowcliffe's R-package "activity" [20].

To investigate whether fusing temporal information can improve wildlife recognition performance, we designed a neural network for combining wildlife image features and temporal features, named Temporal-SE-ResNet50. First, we utilized the ResNet50 model and introduced SE attention to extract wildlife image features. Then, we gained temporal metadata from wildlife images. After applying cyclical encoding, which uses sine–cosine mapping for handling periodic data such as the date and time, the temporal features were then extracted by a residual multilayer perceptron (MLP) network. Finally, the wildlife image features and temporal features were fused by a dynamic MLP module to obtain the final recognition results.

Our contribution includes the following three parts:

- We utilized temporal metadata in camera trap images to aid wildlife recognition and found that extracting temporal features after cyclical encoding of the date and time, respectively, can effectively improve the accuracy of wildlife recognition, which provides a new idea for using animal domain knowledge like animal rhythms to improve wildlife image recognition;
- We proposed a wildlife recognition framework called Temporal-SE-ResNet50 that fuses image features and temporal features, which uses an SE-ResNet50 network to extract image features, a residual MLP network to extract temporal features, and then uses a dynamic MLP module to fuse the above features together;
- We conducted extensive experiments on three national park camera trap datasets and demonstrated that our method is effective in improving wildlife recognition performance.

The remainder of this paper is organized as follows: Section 2 describes the framework used for fusing image features and temporal features, including SE-ResNet50 for extracting image features, a residual MLP network for extracting temporal features, and a dynamic MLP module for fusing image features and temporal features. Section 3 describes the data sources, experimental setup, and evaluation metrics. Section 4 discusses the experimental

results. Section 5 discusses the experimental results and future research directions. Section 6 presents the conclusions.

2. Methods

In this section, we introduce Temporal-SE-ResNet50, a wildlife recognition framework that fuses image and temporal information. As shown in Figure 2, the overall framework of Temporal-SE-ResNet50 consists of four stages. In the image feature extraction stage, we construct SE-ResNet50 to extract wildlife features from images more effectively than ResNet50 (see Section 2.1); in the temporal metadata acquisition stage, we first obtain the temporal metadata from each camera trap image; in the temporal feature extraction stage, we obtain the corresponding temporal features of the image through the residual MLP network (see Section 2.2); and in the image feature and temporal feature fusion stage, after obtaining the image features and temporal features, we use the dynamic MLP module to fuse the two to obtain the enhanced image representation (see Section 2.3). In the end, we obtain the recognition results on different species of wildlife.

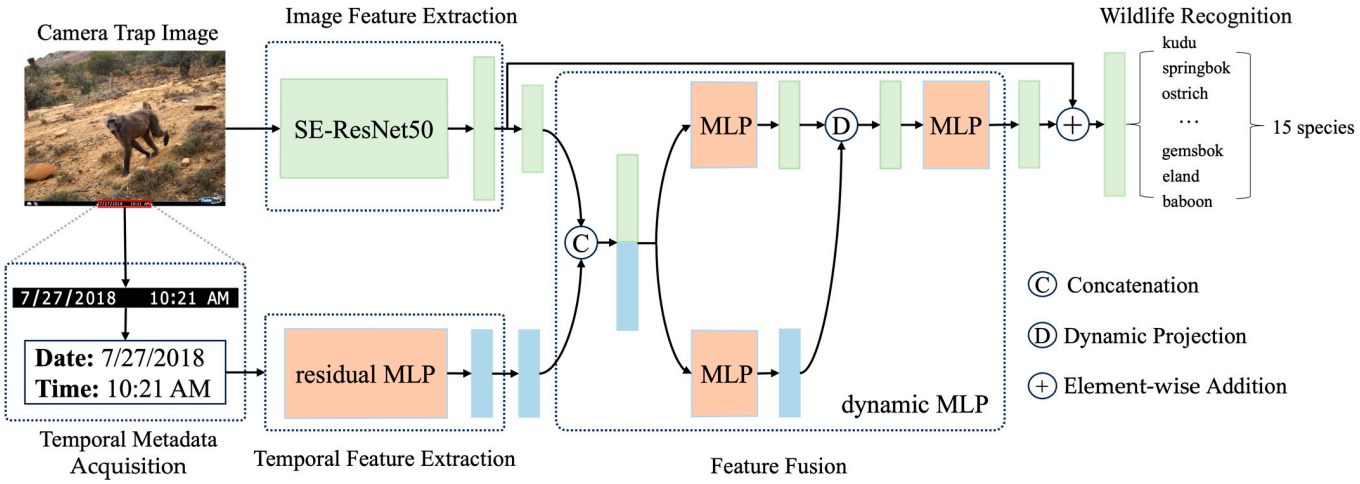


Figure 2. Overall framework of Temporal-SE-ResNet50. The framework is mainly divided into four parts: image feature extraction, temporal metadata acquisition, temporal feature extraction, and feature fusion.

2.1. Camera Trap Image Feature Extraction

Wildlife images obtained by utilizing camera traps in natural scenes are usually affected by lighting conditions, animal behaviors, shooting angles, backgrounds, etc., making recognition challenging. Therefore, we designed the SE-ResNet50 model based on ResNet [21] and squeeze-and-excitation network (SENet) [22] to extract wildlife image features. The structure of the SE-ResNet50 model is shown in Figure 3.

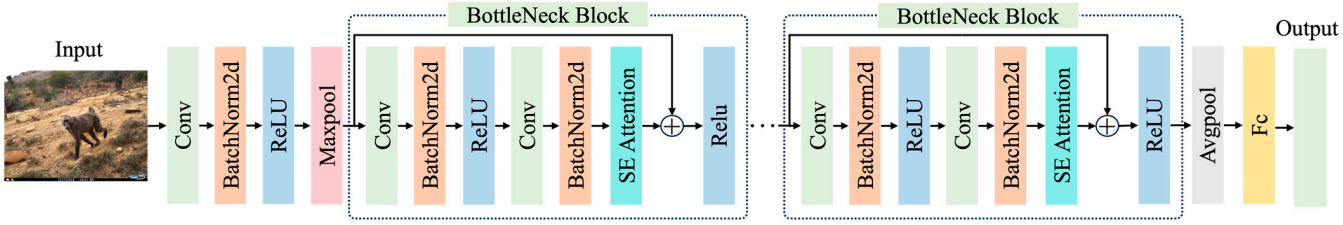


Figure 3. Overall structure of the SE-ResNet50 model. The model uses the ResNet50 network to extract wildlife features from camera trap images and adds SE attention to suppress unimportant features.

ResNet has had excellent performance in numerous previous wildlife image recognition studies [6,8]. Given the computation and network complexity, we chose a 50-layer ResNet as the basic network. ResNet-50 starts with a regular convolutional layer for the

initial feature extraction from input images. It is then followed by four residual blocks. As shown in Figure 3, each residual block consists of multiple stacked BottleNeck blocks, where each BottleNeck block typically incorporates multiple convolutional layers, which help the model to extract different animal features from the input data; e.g., the shallow residual block extracts simple features such as contours, and the deep residual block extracts detailed features such as tails and hairs. To maintain the integrity of the information and to address the problem of gradient degradation during model training, skip connections are used in every BottleNeck block. These connections allow input features to be added directly to the output, ensuring that key details are preserved during the learning process. Subsequently, global average pooling is applied to convert the feature map into a fixed-length representation. This step helps to summarize the important features of the data over the entire spatial dimension. Finally, the resulting fixed-length vector representation is passed to the fully connected layer, which is responsible for performing specific classification tasks.

Recently, a large number of studies [23,24] have demonstrated that adding an attention mechanism can effectively enhance the ability of convolutional neural networks to extract key features. Given that camera trap images usually have complex backgrounds and different lighting conditions, in order to further enable the model to better focus on key animal regions and ignore other irrelevant information, we introduce the SE attention module. This module uses global pooling to compress the global spatial information and then learns the importance of each channel in the channel dimension. As shown in Figure 4, it is specifically divided into three operations: squeeze, excitation, and reweight.

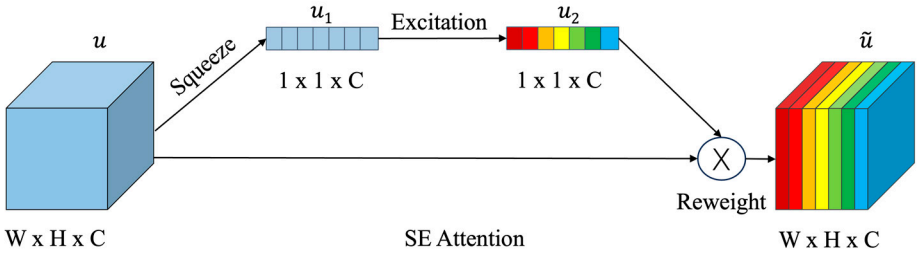


Figure 4. Overall structure of the SE attention. The structure contains three parts: Squeeze, Excitation, and Reweight. With this module, wildlife features can be extracted better from images.

The squeeze operation compresses the input feature map $u \in R_{W\times H\times C}$ through the spatial dimension using global average pooling and obtains the feature map $u_1 \in R_{1\times 1\times C}$, which represents the global distribution of the responses on the feature channels.

$$u_1 = F_{squeeze}(u) = \frac{1}{W \times H}\sum_{i=1}^{W}\sum_{j=1}^{H} u(i,j). \tag{1}$$

The excitation operation models the correlation between the feature channels significantly by using two fully connected layers. To reduce the computational effort, the first fully connected layer compresses the channel C with a ratio of r, and the second fully connected layer then restores it to C. The feature map $u_2 \in R_{1\times 1\times C}$ with channel weights is then obtained using Sigmoid activation.

$$u_1' = \delta(W_1, u_1), \tag{2}$$

$$u_2 = F_{excitation}(u_1,\ W) = \sigma\left(W_2 u_1'\right), \tag{3}$$

where σ denotes the Sigmoid activation function, $W_2 \in R^{C\times\frac{C}{r}}$, δ denotes the ReLU activate function, and $W_1 \in R^{\frac{C}{r}\times C}$.

The reweight operation obtains the final feature map $\widetilde{u} \in R_{W \times H \times C}$ by the multiplication channel weights with the original input feature map.

$$\widetilde{u} = F_{reweight}(u, u_2) = u \cdot u_2. \tag{4}$$

We introduced SE attention module in each residual structure and constructed the SE-ResNet50 model, which can pay more attention to the key wildlife features and suppress the other unimportant features.

2.2. Temporal Feature Extraction

Since dates have a cyclical feature, the end of one year and the start of the next year should be close to each other. Therefore, we use sine–cosine mapping [16] to encode the date metadata captured by the camera trap as (d_1, d_2) according to Equations (5) and (6). With this cyclical encoding, December 31st and January 1st are mapped to be near each other.

$$d_1 = sin\left(\frac{2\pi d}{365}\right), \tag{5}$$

$$d_2 = cos\left(\frac{2\pi d}{365}\right), \tag{6}$$

where 365 respects the total number of days in a year, or 366 in a leap year.

Correspondingly, for the time metadata, we also use the sine–cosine mapping to encode the time metadata captured by the camera trap as (t_1, t_2) according to Equations (7) and (8). With this cyclical encoding, 23:59 and 0:00 were also mapped to be near each other.

$$t_1 = sin\left(\frac{2\pi t}{1440}\right), \tag{7}$$

$$t_2 = cos\left(\frac{2\pi t}{1440}\right), \tag{8}$$

where 1440 respects the total number of minutes in a day.

Finally, the date and time metadata are combined to form our temporal information (d_1, d_2, t_1, t_2).

Inspired by PriorsNet [25], we extract temporal features from a residual MLP network. As shown in Figure 5, it is a fully connected neural network model that consists of two fully connected layers and four fully connected residual layers. Each fully connected residual layer contains two fully connected layers, two ReLU activation functions, and a Dropout layer composition. By inputting temporal information into this network model, the feature mapping of the temporal information is finally obtained. The dimension of the temporal features is set to 256, which achieves the best performance, as described by Tang et al. [26].

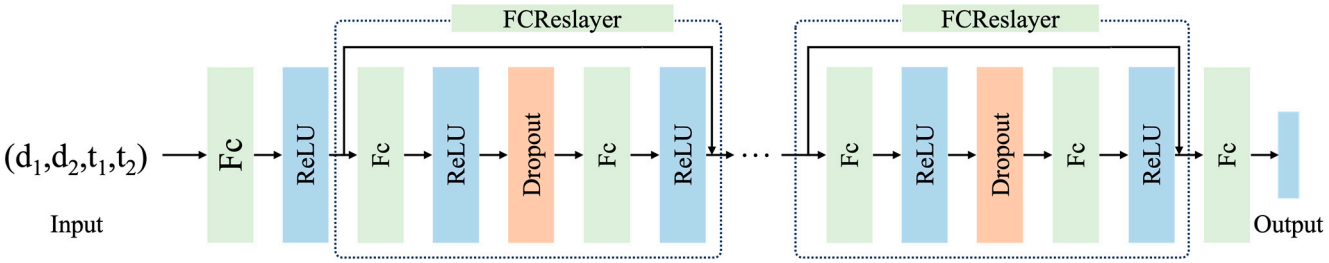


Figure 5. Overall structure of the residual MLP network. The temporal features are obtained by taking the cyclically encoded temporal information and feeding it into this network.

2.3. Image Feature and Temporal Feature Fusion

After the image features and temporal features were obtained, we used the dynamic MLP module [27] to fuse the wildlife image features and temporal features, and the overall

structure of the fusion is shown in Figure 6. Since the dimensionality of the image features is much more than that of the temporal features, for better feature fusion, we first performed a dimensionality reduction operation; i.e., we reduced the dimensionality of the image features to 256, which exists only in Temporal-ResNet50 and Temporal-SE-ResNet50.

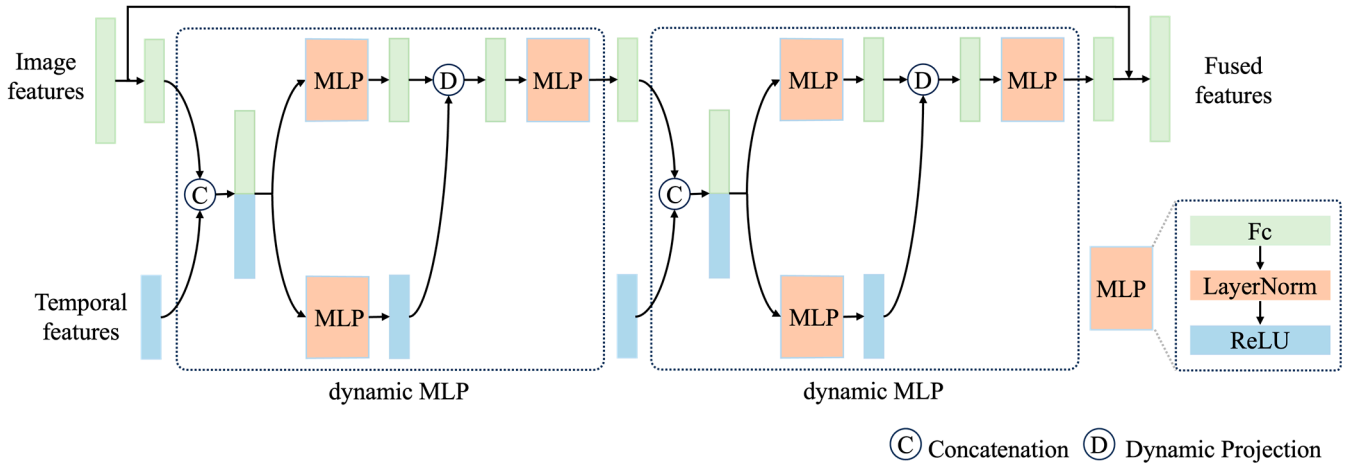


Figure 6. Overall structure of the fusion module. Each MLP block consists of a fully connected layer, a layer normalization, and a ReLU activation function.

The image features extracted by the convolutional neural network are three-dimensional in shape as (H, W, C), where H and W denote the height and width of the feature map, respectively, and C denotes the number of channels. And the temporal features are one-dimensional. When splicing was performed, the temporal features were mapped to the same dimension as the image features; then, the text features and image features were concatenated in accordance with the channel direction, and then feature fusion was performed by a convolutional neural network. So, in a dynamic MLP module, image features and temporal features were first concatenated together channel-wise and then fed into the subsequent MLP block to generate image features and temporal features, respectively. Guided by the temporal features, projection parameters were dynamically generated to adaptively improve the representation of image features. Finally, after the MLP block, we obtained the fused features.

After stacking two dynamic MLP modules, we obtained the fused wildlife features. We expanded the dimensions to the original output image feature dimension. By using a skip connection, we used the fused features to further enhance the representation of the original image features. Finally, after the fully connected layer, the learned features were mapped to the output categories to obtain the final recognition result.

3. Experiments

3.1. Camdeboo DataSet

The data used in this paper came from the Snapshot Camdeboo project, part of the Snapshot Safari network. Images were collected from camera traps set in the Camdeboo National Park of South Africa from July 2018 to February 2019, totaling 12,132 camera trap image sequences. Considering the purpose of our study, we removed the image sequences labeled human and null from the original camera trap image sequences, and we ended up with 9859 images of 15 animals. Figure 7 shows some examples of images from our dataset.

We obtained the date and time of observation from the bottom of each image. Given that the capture time metadata from the camera trap images were in 12 h format, we converted the time metadata to 24 h format.

To ensure that our model can be better adapted to national park monitoring scenarios, it is necessary to maintain a consistent distribution of the partitioned training and test set categories, so we used random sampling for each species. Finally, our dataset was divided

into a 75% training set and a 25% test set, corresponding to 7397 images and 2462 images, respectively. Table 1 shows details of the Camdeboo dataset.

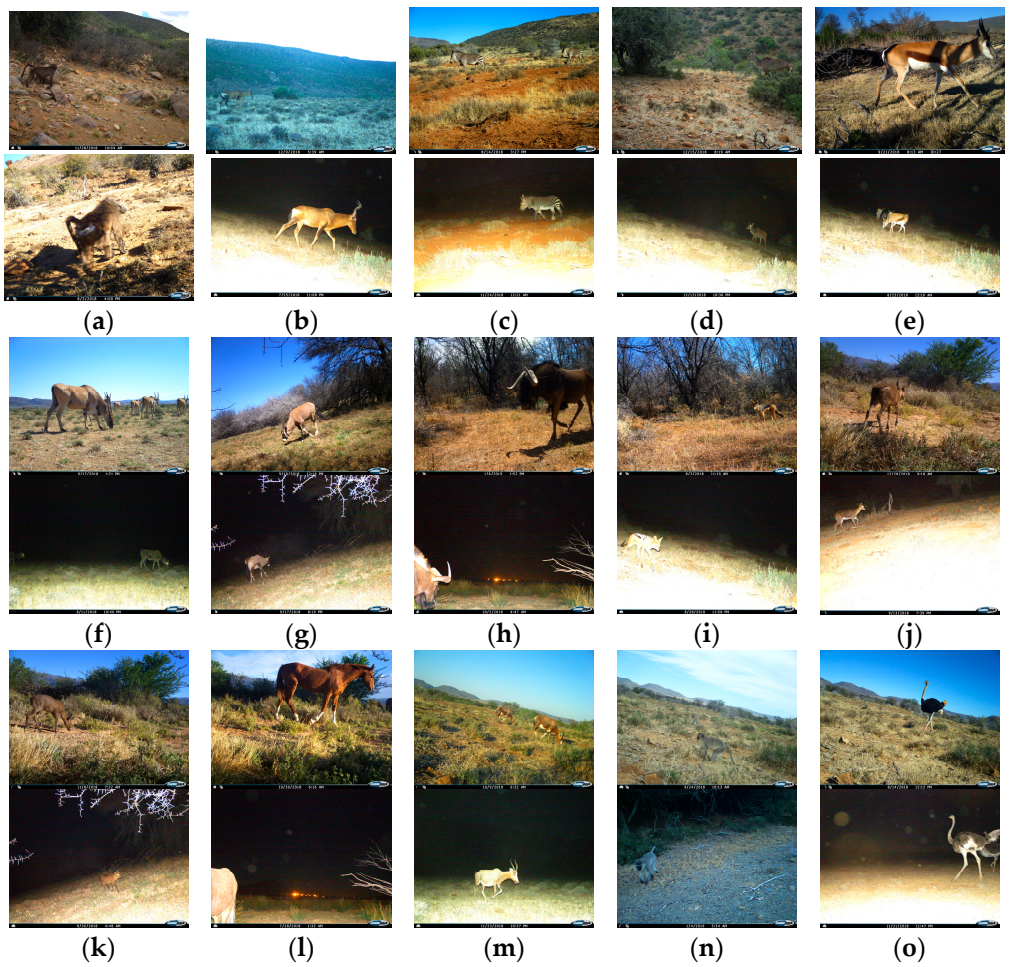


Figure 7. Some examples of images from the Camdeboo dataset spanning different times of the day. (**a**) Baboon. (**b**) Red hartebeest. (**c**) Mountain zebra. (**d**) Kudu. (**e**) Springbok. (**f**) Eland. (**g**) Gemsbok. (**h**) Black wildebeest. (**i**) Black-backed jackal. (**j**) Mountain reedbuck. (**k**) Grey duiker. (**l**) Horse. (**m**) Blesbok. (**n**) Vervet monkey. (**o**) Ostrich.

Table 1. Details of the Camdeboo dataset.

Species	Number of Training Images	Number of Test Images
kudu	1614	538
springbok	1033	344
ostrich	979	326
blesbok	586	195
red hartebeest	507	169
vervet monkey	495	165
mountain zebra	486	162
gemsbok	452	150
grey duiker	418	139
eland	378	126
black wildebeest	144	48
black-backed jackal	122	40
horse	85	28
mountain reedbuck	55	18
baboon	43	14

3.2. Other Camera Trap Datasets

We also obtained two camera trap datasets from other national parks in Africa, including a subset of the Snapshot Serengeti dataset [28] and the Snapshot Mountain Zebra dataset, which is part of the Snapshot Safari network. This subset of the Snapshot Serengeti dataset contains 400,000 images, most of which are empty. After removing the images labeled as blank, human, and other less numerous species, 57,588 images of 20 species were obtained. Out of these, 75% of the images were used as a training set and 25% were used as a test set. The details of the Snapshot Serengeti dataset are shown in Table 2.

Table 2. Details of the Snapshot Serengeti dataset.

Species	Number of Training Images	Number of Test Images	Species	Number of Training Images	Number of Test Images
Thomson's gazelle	21,754	7263	dik-dik	543	182
zebra	3990	1339	impala	515	172
Grant's gazelle	3537	1178	wildebeest	482	158
guinea fowl	2371	789	cheetah	425	140
warthog	1916	638	kori bustard	399	128
hartebeest	1517	513	topi	291	99
giraffe	1408	460	reptiles	264	89
elephant	1253	417	baboon	257	84
buffalo	1075	359	hare	244	81
reedbuck	709	236	bat-eared fox	235	76

The Snapshot Mountain Zebra dataset was collected in Mountain Zebra National Park, which is located in the Eastern Cape of South Africa. For the Snapshot Mountain Zebra dataset, we performed the same cleaning operation and obtained 4753 wildlife images of 10 species. After splitting the data for the training and test sets, the details of the Snapshot Mountain Zebra dataset are shown in Table 3.

Table 3. Details of the Snapshot Mountain Zebra dataset.

Species	Number of Training Images	Number of Test Images
mountain zebra	1416	470
kudu	407	136
springbok	332	110
black wildebeest	265	88
red hartebeest	255	84
baboon	224	75
black-backed jackal	210	70
vervet monkey	192	64
buffalo	145	48
eland	122	40

3.3. Experiment Settings

All experiments were performed on Ubuntu 16.04 and NVIDIA GeForce RTX 2080Ti GPU. Table 4 shows the configuration of software and hardware used in all experiments.

Table 4. Experiment configuration of the software and hardware.

Environment	Configuration Properties
System	Ubuntu 16.04
GPU	NVIDIA GeForce RTX 2080Ti
CPU	Intel(R) Xeon(R) Platinum 8255C CPU
Development Framework	PyTorch 1.7.1
Programming Language	Python 3.6

All camera trap images were resized to 256 × 256. During training, data enhancement methods included image random crop to 224 × 224, random horizontal flip, and Mixup [29]. We trained the proposed model using the label-smoothing loss function and set the maximum number of epochs to 90. We used the SGD optimizer with momentum 0.9 and weight decay 1×10^{-4}. The initial learning rate was set to 0.01 and the batch size was set to 64. We adopted the warmup strategy, and the learning rate adjustment method was defined as follows:

$$lr = lr_{init} \times \frac{1}{2}\left(1 + cos\left(\frac{1 + \pi(epoch_{current} + 1)}{epoch_{max}}\right)\right). \quad (9)$$

3.4. Evaluation Metrics

To evaluate the performance of the proposed network, we used accuracy as the evaluation metric for the model. Accuracy is defined as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \quad (10)$$

where TP is the total number of true-positive wildlife samples, TN is the total number of true-negative wildlife samples, FP is the total number of false-positive wildlife samples, and FN is the total number of false-negative wildlife samples.

4. Results

In this section, we demonstrate the efficiency of our proposed method by outlining the outcomes of six different experiments. To begin with, we determined the optimal learning rate and batch size of the proposed model by performing cross-validation on the training set. Next, we compared our proposed method with other advanced classification methods and demonstrated that our proposed method has better performance in wildlife identification. Additionally, we compared the recognition accuracy of ResNet50 and Temporal-SE-ResNet50 on different species. Moreover, we conducted a case study to analyze the reasons for the decline in the recognition accuracy of certain species. Furthermore, we performed an ablation analysis of the proposed method to prove the importance of each module. In addition, we compared the effects of different temporal coding methods on the model performance. Then, we compared the effects of adding different attention modules on the model performance. Finally, we validated the generality of our proposed method on different national park camera trap datasets, such as the Snapshot Serengeti dataset and the Snapshot Mountain Zebra dataset.

4.1. Comparison Experiments with Different Learning Rates and Batch Sizes on the Camdeboo Dataset

Considering the lack of a uniform standard for setting hyperparameters in wildlife recognition, different learning rates and batch sizes were used on different datasets, and we used the control variable method to study the effect of different learning rates and batch sizes on wildlife recognition on the Camdeboo dataset. The performance of the model under different learning rates and different batch sizes was obtained by using 5-fold cross-validation on the training set. Under the condition of a fixed batch size of 64, we set the initial learning rates to 0.04, 0.03, 0.02, 0.01, and 0.001, respectively, based on the experimental settings of Ding et al. and Lv et al. [30,31]. As shown in Figure 8a, the average accuracy of 5-fold cross-validation on the training set was highest at 91.31% when the learning rate was 0.01. Based on the principle that a batch size of a power of two can better utilize CPU or GPU performance and experimental settings from other studies [30–32], we set the batch sizes to eight, sixteen, thirty-two, sixty-four, and one-hundred and twenty-eight when fixing the initial learning rate to 0.01. As shown in Figure 8b, the average accuracy of 5-fold cross-validation on the training set was up to 91.31% when the batch

size was 64. Therefore, we chose the batch size 64 and learning rate 0.01 as our optimal hyperparameters and then retrained our model.

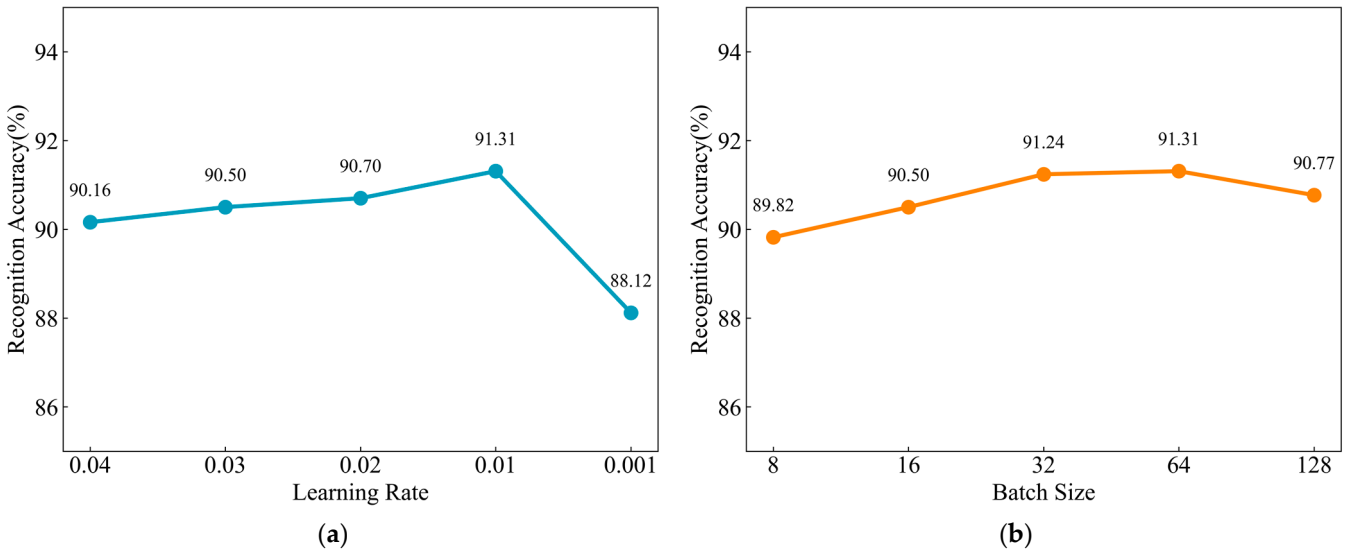


Figure 8. The choice of hyperparameters for our proposed model. (**a**) Comparison experiments with different learning rates. (**b**) Comparison experiments with different batch sizes.

4.2. Comparison Experiments with Other Advanced Classification Methods on the Camdeboo Dataset

To evaluate the performance of our proposed network model, we conducted comparative experiments with commonly used convolutional neural network models on the Camdeboo dataset, such as VGGNet [33], ResNet50, EfficientNet [34], ShuffleNetV2 [35], MobileNetV2 [36], MobileNetV3 [37], and ConvNeXt [38]. All models used pre-trained weights on ImageNet and then were fine-tuned on our dataset. As shown in Table 5, our proposed Temporal-SE-ResNet50 model achieved the highest accuracy of 93.10%, while the EfficientNet-B0 model had the lowest accuracy of only 77.90%. The MobileNetV3-L model achieved the highest accuracy in the MobileNet family at 90.17%, and the ShuffleNetV2-2.0x model achieved the highest accuracy in the ShuffleNet family at 91.75%. Compared with the ResNet50, VGG19, ShuffleNetV2-2.0x, MobileNetV3-L, and ConvNeXt-B models, the accuracy of our method improved by 0.53%, 0.94%, 1.35%, 2.93%, and 5.98%, respectively.

Table 5. Accuracy of different classification methods on the Camdeboo dataset.

Method	Batch Size	Learning Rate	Epoch	Accuracy
MobileNetV2 [36]	64	0.01	90	89.24%
MobileNetV3-S [37]	64	0.01	90	86.35%
MobileNetV3-L [37]	64	0.01	90	90.17%
EfficientNet-B0 [34]	64	0.01	90	77.90%
ShuffleNetV2-0.5x [35]	64	0.01	90	78.02%
ShuffleNetV2-1.0x [35]	64	0.01	90	86.31%
ShuffleNetV2-1.5x [35]	64	0.01	90	90.45%
ShuffleNetV2-2.0x [35]	64	0.01	90	91.75%
VGG19 [33]	64	0.01	90	92.16%
ResNet50 [21]	64	0.01	90	92.57%
ConvNeXt-T [38]	64	0.01	90	80.63%
ConvNeXt-S [38]	64	0.01	90	80.46%
ConvNeXt-B [38]	64	0.01	90	87.12%
Temporal-SE-ResNet50	64	0.01	90	**93.10%**

Furthermore, we compared ResNet50 and Temporal-SE-ResNet50 in terms of the training time and average recognition time for a single wildlife image. As shown in Table 6, compared to ResNet50, our proposed method took 1 min and 53 s longer in training time, which is perfectly acceptable compared to the improved accuracy.

Table 6. Comparison of different models in terms of training time on the Camdeboo dataset.

Method	Training Time
ResNet50 [21]	2 h 37 min 16 s
Temporal-SE-ResNet50	2 h 39 min 9 s

4.3. Comparative Experiments on Recognition Accuracy of Different Species on the Camdeboo Dataset

To further investigate the effect of temporal features on the species recognition performance, we compared the recognition accuracy of each species in the Camdeboo dataset on ResNet50 and Temporal-SE-ResNet50, respectively. As shown in Figure 9a,b, the recognition accuracy of eight species, including springbok, ostrich, blesbok, mountain zebra, gemsbok, eland, black-backed jackal, and mountain reedbuck, increased by 3.16% on average, with the lowest improvement of 0.62% for ostrich and the highest improvement of 11.11% for mountain reedbuck. The recognition accuracy of five species, including kudu, vervet monkey, grey duiker, horse, and baboon, decreased by 2.70% on average, the lowest decrease in recognition accuracy was 0.61% for vervet monkey, and the highest decrease in recognition accuracy was 7.14% for baboon.

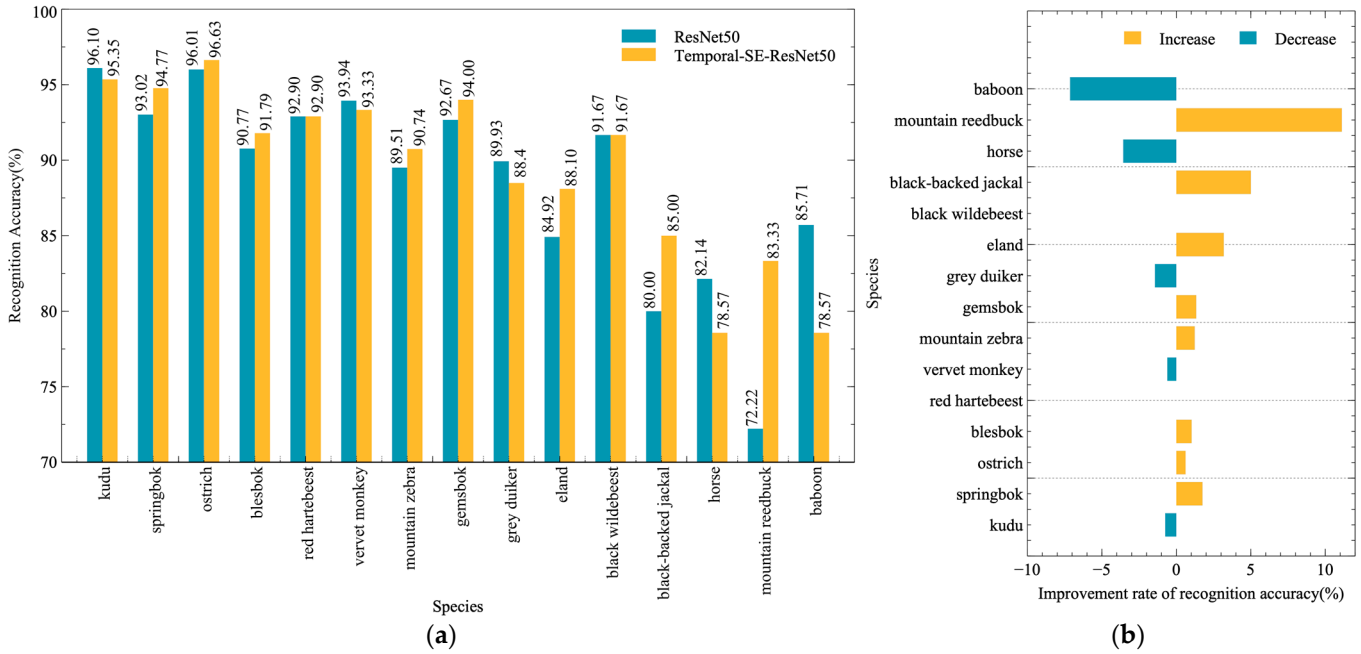


Figure 9. Recognition accuracy for each species on the Camdeboo dataset. (**a**) Recognition accuracy for each species on ResNet50 and Temporal-SE-ResNet50. (**b**) Improvement rate in recognition accuracy of our method compared to ResNet50 for each species.

As shown in Figure 10, we computed confusion matrices to investigate the effect of our method on the recognition performance of different species. In the confusion matrix, the larger the value of the diagonal elements, the better; and the smaller the value of the elements at other positions, the better. Comparing Figure 10a,b, for springbok, compared to ResNet50, our approach mainly reduces the percentage of its misclassification as blesbok, thus improving the recognition accuracy of springbok.

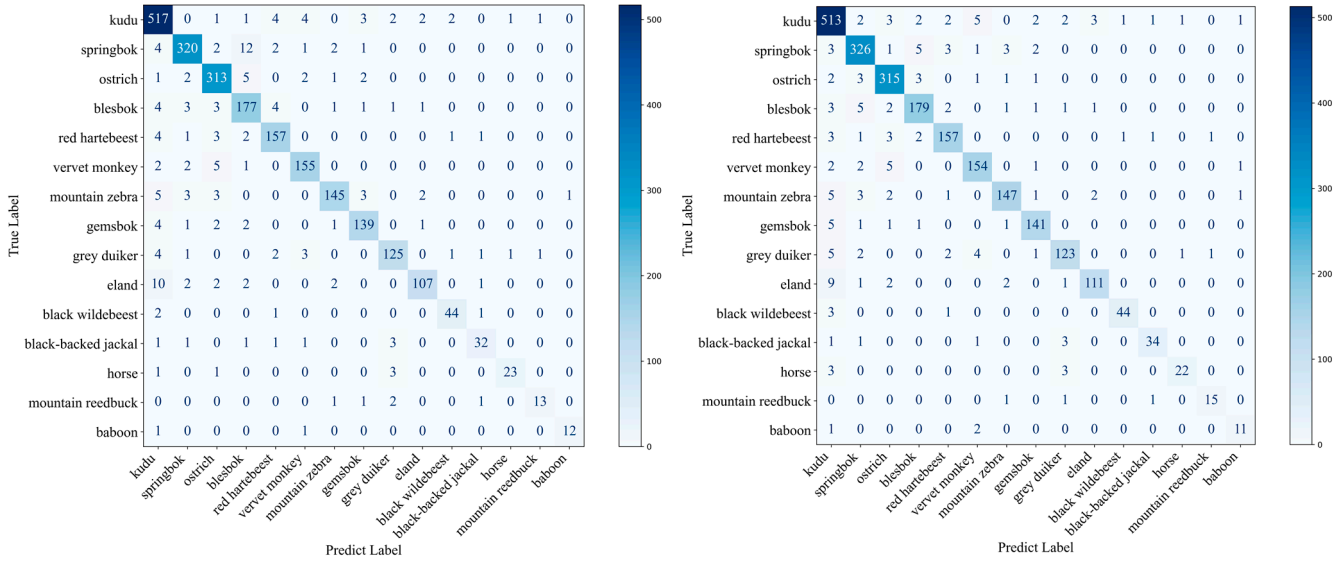


Figure 10. Confusion matrix. (**a**) ResNet50. (**b**) Temporal-SE-ResNet50.

4.4. Case Study

From Figures 9 and 10, we found that the decrease in recognition accuracy was mainly concentrated in baboon and horse images, partially focused on the grey duiker, kudu, and vervet monkey. As shown in Figure 11, it is believed that this decline was due to the following two factors. On the one hand, it is believed that this decrease was related to the rarity of the species. The small number of the species resulted in a relatively small number of valid images collected by the camera trap, and the model was unable to learn patterns of animal movement from the small amount of data, thus leading to a decrease in recognition accuracy for species such as baboons and horses. On the other hand, for relatively large species such as the grey duiker, kudu, and vervet monkey, because the number of images collected by camera traps in different months was relatively uniform, the accuracy of species recognition could not be improved after fusing the temporal features.

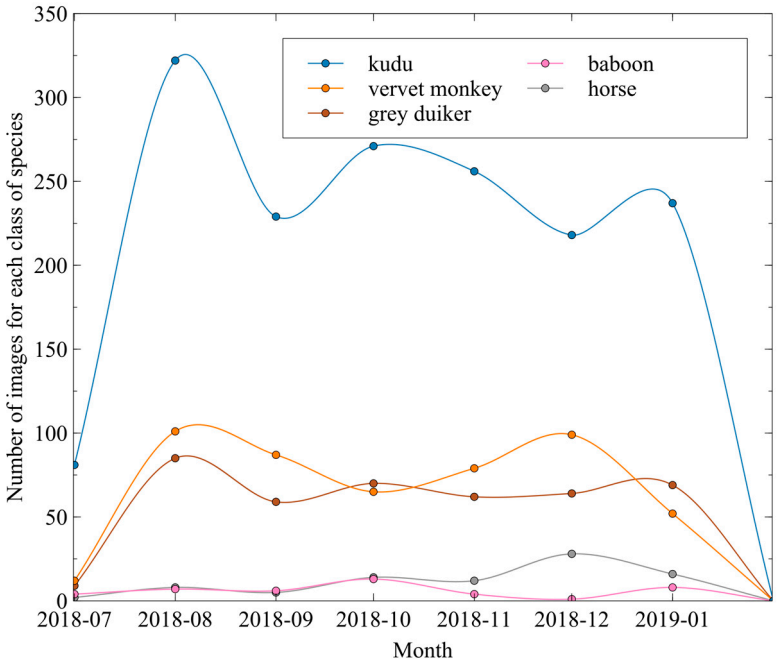


Figure 11. Distribution of the number of camera trap images collected in different months.

4.5. Ablation Studies

We conducted ablation experiments on the Camdeboo dataset from different perspectives, including the importance of each module, the importance of different temporal encoding styles, and the importance of different attentional modules.

4.5.1. The Importance of Different Modules in Our Proposed Method

To investigate the contributions of each module to our proposed method, we performed ablation experiments. With the ResNet50 model as a baseline, we considered three scenarios: adding only temporal information, adding only the attention module, and adding both temporal information and the attention module. Table 7 shows that only fusing the temporal information or only adding the attention module to the baseline model improved the accuracy of wildlife recognition by 0.44% and 0.28%, respectively. When both fusing the temporal information and adding the attention module, the accuracy increased by 0.53%.

Table 7. Ablation study of temporal information and attention module.

Model	Temporal Information	Attention	Accuracy
Baseline	✗	✗	92.57%
	✓	✗	93.01%
	✗	✓	92.85%
Temporal-SE-ResNet50	✓	✓	**93.10%**

4.5.2. The Effect of Different Temporal Encoding Methods

To find the optimal temporal encoding method, we compared the effects of different temporal coding methods on the wildlife recognition model when not embedded in the SE attention module. The first way was to fit the date and time separately to [−1,1] and then join them together. The second and third ways were to cyclically encode the time and date individually using sine–cosine mapping. The last way was to cyclically encode the date and time using sine–cosine mapping and then join them together. Table 8 shows that when both the date and time were periodically encoded and then fused with image features, better accuracy was obtained, with an improvement of about 0.44%.

Table 8. Accuracy of test set under different temporal information encoding methods.

Model	Accuracy
ResNet50	92.57%
+ Temporal information without cyclical encoding	92.61%
+ Only date with cyclical encoding	92.65%
+ Only time with cyclical encoding	92.57%
+ Temporal information with cyclical encoding	**93.01%**

4.5.3. The Importance of the Attention Module

Since the attention mechanism [39] is commonly used in visual tasks to enhance the feature extraction capability of the network, we considered whether it could be added to our model to further improve performance. We embedded different attention modules in the same position of the BottleNeck block of the ResNet50 model to find the suitable module for the added attention. As shown in Table 9, the recognition accuracy decreased by 0.4%, 0.28%, and 0.24% when adding the CBAM [40], ECA [23], or CA [24] attention modules, respectively. Only by embedding the SE attention module did the recognition accuracy further increase to 93.10%.

Table 9. Comparison of different attention modules when epoch is 90, batch size is 64, and learning rate is 0.01.

Model	Accuracy
Temporal-ResNet50	93.01%
+ CBAM [40]	92.61%
+ ECA [23]	92.73%
+ CA [24]	92.77%
+ SE [22] (Temporal-SE-ResNet50)	**93.10%**

4.6. Comparative Experiments on Different Wildlife Camera Trap Datasets

To further demonstrate the effectiveness of the proposed method, we executed experiments on a subset of the Snapshot Serengeti dataset and the Snapshot Mountain Zebra dataset. We followed the experimental setup in Section 3.2 and compared the recognition accuracy of our proposed method with that of the baseline method.

The experimental results are shown in Table 10, where our proposed method's accuracy was improved by 0.25% and 0.42% over the baseline method's accuracy on the Snapshot Serengeti dataset and the Snapshot Mountain Zebra dataset, respectively. This proved that our proposed method was general and could work on different camera trap datasets.

Table 10. Accuracy of different classification methods on the Snapshot Serengeti dataset and the Serengeti Mountain Zebra dataset when epoch is 90, batch size is 64, and learning rate is 0.01.

Dataset	Method	Accuracy
Snapshot Serengeti	ResNet50 [21]	95.92%
	Temporal-SE-ResNet50	**96.17%**
Snapshot Mountain Zebra	ResNet50 [21]	89.03%
	Temporal-SE-ResNet50	**89.45%**

5. Discussion

In this study, we present a novel method that fuses image and temporal metadata for recognizing wildlife in camera trap images. Our experimental results on different camera trap datasets demonstrate that leveraging temporal metadata can improve overall wildlife recognition performance, which is similar to the findings of Terry et al. [15] and de Lutio et al. [16] who utilized contextual data to enhance the recognition performance of citizen science images.

In recent years, a large number of wildlife recognition studies based on camera trap images have achieved great accomplishments [6–8]. These achievements are attributed to large amounts of labeled data. To further improve wildlife recognition performance, the first step is to expand the dataset. However, the process of data annotation is time-consuming and labor-intensive. To solve this problem without expanding the dataset, we consider utilizing information from existing images, such as the observation times of camera traps. By analyzing the frequency and distribution of random animal encounters in camera traps, this can be used not only to estimate animal population density [18,19], but also to reflect animal species activity patterns [41]. Unlike previous wildlife recognition studies which were merely based on camera trap images, we exploit the observation time of the camera trap images to help interpret the images.

In addition to wildlife recognition using camera traps, species recognition using citizen science images is also a hot research topic. There are many studies that utilize geographic and contextual metadata to aid species recognition from public science images. Chu et al. [42] used geolocation for fine-grained species identification, which improved the top-1 accuracy in iNaturalist from 70.1% to 79.0%. Mac Aodha et al. [25] proposed a method to estimate the probability of species occurrence at a given location using the geographic location and time as a priori knowledge. However, studies utilizing the above information in camera trap data are rare. Consistent with previous studies on citizen science images

that utilize geographic and contextual metadata, our findings indicated that fusing the temporal information enhanced the baseline accuracy by 0.44%. Unlike these studies, we only considered temporal information and did not utilize geographic data. For the temporal metadata, we further considered date and time separately, where date corresponds to the animal's seasonal rhythms and time corresponds to the animal's circadian rhythms. Our experimental results show that the periodic coding of date and time separately, followed by feature extraction and then fusion with image features, can better leverage temporal metadata. As for geographic metadata, geographic data is not easily accessible, and due to the deployment of infrared cameras in a national park with very little geographic variation, the potential for integrating geo-metadata requires further research.

Attention mechanisms have been widely used in the field of animal detection and recognition owing to their plug-and-play and effective traits. Xie et al. [39] introduced SE attention in YOLOv5 to improve big mammal species detection from a UAV viewpoint. Zhang et al. [43] introduced a coordinated attention (CA) module in YOLOv5s to suppress non-critical information on the face of sheep to recognize the identity of sheep in real time. Given the different lighting conditions and backgrounds present in camera trap images, to better extract wildlife features, we compared the performance of the model after adding different attention modules and found that the addition of the SE attention mechanism can further improve wildlife recognition by 0.09%.

In general, our proposed method for wildlife recognition has significant advantages, improving the baseline model by 0.53% on the Camdeboo dataset without additional image data. In addition, on the Snapshot Serengeti dataset and the Snapshot Mountain Zebra dataset, our method improved by 0.25% and 0.43% compared to the baseline model. These findings have important implications for utilizing the potential of animal rhythms for wildlife identification based on camera trap images. Moreover, these temporal metadata are already collected with the camera trap image and do not add an additional collection burden. However, the use of temporal information is less useful for animals with insufficiently trained images. Given that factors affecting wildlife recognition include, in addition to the number of training images, life habits (e.g., the presence of one or more identical wildlife on a single image as a result of living in a group or solitary), motion poses (e.g., running results in the presence of blurring in the image), and the timing of the shot (e.g., the image contains only a portion of the animal's body), determining the minimum amount of training data required for each animal is relatively complex, which deserves to be investigated in depth in future work.

In the future, we will focus on the following two aspects of work. On the one hand, to further explore the potential of fusing temporal information, we will further determine the minimum number of images of each wildlife species to be used in training. On the other hand, considering that animals living in urban areas can experience changes in activity patterns due to artificial light, food availability, and human activities, we will collect datasets of wildlife camera trap images involving different environments or contexts to investigate the robustness of the recognition model after fusing the temporal metadata under different activity patterns.

6. Conclusions

In the paper, we have proposed a Temporal-SE-ResNet50 network model that fuses trap camera images and additional temporal metadata acquired in the images for wildlife recognition. The model constructs SE-ResNet50 based on ResNet50 and the SE attention module to extract wildlife features from camera trap images as well as a residual MLP network to extract temporal features and then fuses the image features and temporal features through a dynamic MLP module for the final wildlife recognition. The experimental results on the Camdeboo dataset show that the recognition accuracy of this method is 93.10%, which is superior to the method of wildlife recognition using only images. We also demonstrate the effectiveness of our method on other camera trap datasets. Our research results have important implications for the use of temporal metadata for wildlife recognition.

Author Contributions: L.L. conceived and conducted the study, including wildlife dataset construction, methodology proposal, experimental design, and writing of the first draft of the paper. C.M. and F.X. supervised the study while reviewing and editing the writing. All authors have read and agreed to the published version of the manuscript.

Funding: This research was jointly funded by the National Key R&D Program of China (2022YFF1302700), the Emergency Open Competition Project of National Forestry and Grassland Administration (202303), and Outstanding Youth Team Project of Central Universities (QNTD202308).

Institutional Review Board Statement: Not applicable.

Data Availability Statement: Snapshot Camdeboo dataset is available at https://lila.science/datasets/snapshot-camdeboo (accessed on 19 February 2024). Snapshot Serengeti dataset is available at https://lila.science/datasets/snapshot-serengeti (accessed on 19 February 2024). Snapshot Mountain Zebra dataset is available at https://lila.science/datasets/snapshot-mountain-zebra (accessed on 19 February 2024). The protocol used for all datasets is the Community Data License Agreement (permissive variant) and this license agreement allows for the free use, modification and distribution of the data.

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

References

1. Brondizio, E.S.; Settele, J.; Díaz, S.; Ngo, H.T. *Global Assessment Report on Biodiversity and Ecosystem Services of the Intergovernmental Science-Policy Platform on Biodiversity and Ecosystem Services*; IPBES secretariat: Bonn, Germany, 2019.
2. Caravaggi, A.; Banks, P.B.; Burton, A.C.; Finlay, C.M.V.; Haswell, P.M.; Hayward, M.W.; Rowcliffe, M.J.; Wood, M.D. A Review of Camera Trapping for Conservation Behaviour Research. *Remote Sens. Ecol. Conserv.* **2017**, *3*, 109–122. [CrossRef]
3. Feng, L.; Zhao, Y.; Sun, Y.; Zhao, W.; Tang, J. Action Recognition Using a Spatial-Temporal Network for Wild Felines. *Animals* **2021**, *11*, 485. [CrossRef]
4. Massei, G.; Coats, J.; Lambert, M.S.; Pietravalle, S.; Gill, R.; Cowan, D. Camera Traps and Activity Signs to Estimate Wild Boar Density and Derive Abundance Indices. *Pest Manag. Sci.* **2018**, *74*, 853–860. [CrossRef]
5. Tuia, D.; Kellenberger, B.; Beery, S.; Costelloe, B.R.; Zuffi, S.; Risse, B.; Mathis, A.; Mathis, M.W.; van Langevelde, F.; Burghardt, T.; et al. Perspectives in Machine Learning for Wildlife Conservation. *Nat. Commun.* **2022**, *13*, 792. [CrossRef]
6. Gomez Villa, A.; Salazar, A.; Vargas, F. Towards Automatic Wild Animal Monitoring: Identification of Animal Species in Camera-Trap Images Using Very Deep Convolutional Neural Networks. *Ecol. Inform.* **2017**, *41*, 24–32. [CrossRef]
7. Zualkernan, I.; Dhou, S.; Judas, J.; Sajun, A.R.; Gomez, B.R.; Hussain, L.A. An IoT System Using Deep Learning to Classify Camera Trap Images on the Edge. *Computers* **2022**, *11*, 13. [CrossRef]
8. Binta Islam, S.; Valles, D.; Hibbitts, T.J.; Ryberg, W.A.; Walkup, D.K.; Forstner, M.R.J. Animal Species Recognition with Deep Convolutional Neural Networks from Ecological Camera Trap Images. *Animals* **2023**, *13*, 1526. [CrossRef] [PubMed]
9. Xie, J.; Li, A.; Zhang, J.; Cheng, Z. An Integrated Wildlife Recognition Model Based on Multi-Branch Aggregation and Squeeze-And-Excitation Network. *Appl. Sci.* **2019**, *9*, 2794. [CrossRef]
10. Yang, W.; Liu, T.; Jiang, P.; Qi, A.; Deng, L.; Liu, Z.; He, Y. A Forest Wildlife Detection Algorithm Based on Improved YOLOv5s. *Animals* **2023**, *13*, 3134. [CrossRef] [PubMed]
11. Zhang, C.; Zhang, J. DJAN: Deep Joint Adaptation Network for Wildlife Image Recognition. *Animals* **2023**, *13*, 3333. [CrossRef]
12. Ahmed, A.; Yousif, H.; Kays, R.; He, Z. Animal Species Classification Using Deep Neural Networks with Noise Labels. *Ecol. Inform.* **2020**, *57*, 101063. [CrossRef]
13. Zhong, Y.; Li, X.; Xie, J.; Zhang, J. A Lightweight Automatic Wildlife Recognition Model Design Method Mitigating Shortcut Learning. *Animals* **2023**, *13*, 838. [CrossRef] [PubMed]
14. Tan, M.; Chao, W.; Cheng, J.-K.; Zhou, M.; Ma, Y.; Jiang, X.; Ge, J.; Yu, L.; Feng, L. Animal Detection and Classification from Camera Trap Images Using Different Mainstream Object Detection Architectures. *Animals* **2022**, *12*, 1976. [CrossRef] [PubMed]
15. Terry, J.C.D.; Roy, H.E.; August, T.A. Thinking like a Naturalist: Enhancing Computer Vision of Citizen Science Images by Harnessing Contextual Data. *Methods Ecol. Evol.* **2020**, *11*, 303–315. [CrossRef]
16. de Lutio, R.; She, Y.; D'Aronco, S.; Russo, S.; Brun, P.; Wegner, J.D.; Schindler, K. Digital Taxonomist: Identifying Plant Species in Community Scientists' Photographs. *ISPRS J. Photogramm. Remote Sens.* **2021**, *182*, 112–121. [CrossRef]
17. Mou, C.; Liang, A.; Hu, C.; Meng, F.; Han, B.; Xu, F. Monitoring Endangered and Rare Wildlife in the Field: A Foundation Deep Learning Model Integrating Human Knowledge for Incremental Recognition with Few Data and Low Cost. *Animals* **2023**, *13*, 3168. [CrossRef]
18. Palencia, P.; Barroso, P.; Vicente, J.; Hofmeester, T.R.; Ferreres, J.; Acevedo, P. Random Encounter Model Is a Reliable Method for Estimating Population Density of Multiple Species Using Camera Traps. *Remote Sens. Ecol. Conserv.* **2022**, *8*, 670–682. [CrossRef]
19. Wearn, O.R.; Bell, T.E.M.; Bolitho, A.; Durrant, J.; Haysom, J.K.; Nijhawan, S.; Thorley, J.; Rowcliffe, J.M. Estimating Animal Density for a Community of Species Using Information Obtained Only from Camera-Traps. *Methods Ecol. Evol.* **2022**, *13*, 2248–2261. [CrossRef]

20. Rowcliffe, J.M.; Kays, R.; Kranstauber, B.; Carbone, C.; Jansen, P.A. Quantifying Levels of Animal Activity Using Camera Trap Data. *Methods Ecol. Evol.* **2014**, *5*, 1170–1179. [CrossRef]
21. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27 June 2016; pp. 770–778.
22. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18 June 2018; pp. 7132–7141.
23. Wang, Q.; Wu, B.; Zhu, P.; Li, P.; Zuo, W.; Hu, Q. ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13 June 2020; pp. 11531–11539.
24. Hou, Q.; Zhou, D.; Feng, J. Coordinate Attention for Efficient Mobile Network Design. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Nashville, TN, USA, 20 June 2021; pp. 13708–13717.
25. Mac Aodha, O.; Cole, E.; Perona, P. Presence-Only Geographical Priors for Fine-Grained Image Classification. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, 27 October 2019, Korea (South); pp. 9596–9606.
26. Tang, K.; Paluri, M.; Fei-Fei, L.; Fergus, R.; Bourdev, L. Improving Image Classification with Location Context. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7 December 2015; pp. 1008–1016.
27. Yang, L.; Li, X.; Song, R.; Zhao, B.; Tao, J.; Zhou, S.; Liang, J.; Yang, J. Dynamic MLP for Fine-Grained Image Classification by Leveraging Geographical and Temporal Information. In Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 18 June 2022; pp. 10935–10944.
28. Swanson, A.; Kosmala, M.; Lintott, C.; Simpson, R.; Smith, A.; Packer, C. Snapshot Serengeti, High-Frequency Annotated Camera Trap Images of 40 Mammalian Species in an African Savanna. *Sci. Data* **2015**, *2*, 150026. [CrossRef]
29. Zhang, H.; Cisse, M.; Dauphin, Y.N.; Lopez-Paz, D. Mixup: Beyond Empirical Risk Minimization. *arXiv* **2018**, arXiv:1710.09412. [CrossRef]
30. Ding, J.; Zhang, C.; Cheng, X.; Yue, Y.; Fan, G.; Wu, Y.; Zhang, Y. Method for Classifying Apple Leaf Diseases Based on Dual Attention and Multi-Scale Feature Extraction. *Agriculture* **2023**, *13*, 940. [CrossRef]
31. Lv, X.; Xia, H.; Li, N.; Li, X.; Lan, R. MFVT: Multilevel Feature Fusion Vision Transformer and RAMix Data Augmentation for Fine-Grained Visual Categorization. *Electronics* **2022**, *11*, 3552. [CrossRef]
32. Chen, R.; Little, R.; Mihaylova, L.; Delahay, R.; Cox, R. Wildlife Surveillance Using Deep Learning Methods. *Ecol. Evol.* **2019**, *9*, 9453–9466. [CrossRef] [PubMed]
33. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv* **2015**, arXiv:1409.1556. [CrossRef]
34. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv* **2017**, arXiv:1704.04861. [CrossRef]
35. Ma, N.; Zhang, X.; Zheng, H.-T.; Sun, J. ShuffleNet V2: Practical Guidelines for Efficient CNN Architecture Design. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8 September 2018; pp. 116–131.
36. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.-C. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18 June 2018; pp. 4510–4520.
37. 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/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October 2019; pp. 1314–1324.
38. Liu, Z.; Mao, H.; Wu, C.-Y.; Feichtenhofer, C.; Darrell, T.; Xie, S. A ConvNet for the 2020s. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 19–20 June 2022; pp. 11976–11986.
39. Xie, Y.; Jiang, J.; Bao, H.; Zhai, P.; Zhao, Y.; Zhou, X.; Jiang, G. Recognition of Big Mammal Species in Airborne Thermal Imaging Based on YOLO V5 Algorithm. *Integr. Zool.* **2023**, *18*, 333–352. [CrossRef]
40. Woo, S.; Park, J.; Lee, J.-Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In *Computer Vision—ECCV 2018*; Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., Eds.; Springer International Publishing: Berlin/Heidelberg, Germany, 2018; pp. 3–19.
41. Hsing, P.-Y.; Hill, R.A.; Smith, G.C.; Bradley, S.; Green, S.E.; Kent, V.T.; Mason, S.S.; Rees, J.; Whittingham, M.J.; Cokill, J.; et al. Large-Scale Mammal Monitoring: The Potential of a Citizen Science Camera-Trapping Project in the United Kingdom. *Ecol. Solut. Evid.* **2022**, *3*, e12180. [CrossRef]
42. Chu, G.; Potetz, B.; Wang, W.; Howard, A.; Song, Y.; Brucher, F.; Leung, T.; Adam, H. Geo-Aware Networks for Fine-Grained Recognition. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), Seoul, Republic of Korea, 27 October 2019; pp. 247–254.
43. Zhang, X.; Xuan, C.; Xue, J.; Chen, B.; Ma, Y. LSR-YOLO: A High-Precision, Lightweight Model for Sheep Face Recognition on the Mobile End. *Animals* **2023**, *13*, 1824. [CrossRef]