*electronics*

MDPI

*Article*

# Enhancing Sika Deer Identification: Integrating CNN-Based Siamese Networks with SVM Classification

**Sandhya Sharma [1,†], Suresh Timilsina [1], Bishnu Prasad Gautam [2], Shinya Watanabe [1], Satoshi Kondo [1] and Kazuhiko Sato [1,*,†]**

[1] Muroran Institute of Technology, 27-1 Mizumoto-cho, Muroran 050-8585, Hokkaido, Japan; 22096503@muroran-it.ac.jp (S.S.); 23096509@muroran-it.ac.jp (S.T.); sin@muroran-it.ac.jp (S.W.); kondo@muroran-it.ac.jp (S.K.)

[2] Department of Information Engineering, Kanazawa Gakuin University, 10 Suemachi, Kanazawa 920-1392, Ishikawa, Japan; gautam@kanazawa-gu.ac.jp

* Correspondence: kazu@muroran-it.ac.jp

† These authors contributed equally to this work.

**Abstract:** Accurately identifying individual wildlife is critical to effective species management and conservation efforts. However, it becomes particularly challenging when distinctive features, such as spot shape and size, serve as primary discriminators, as in the case of Sika deer. To address this challenge, we employed four different Convolutional Neural Network (CNN) base models (EfficientNetB7, VGG19, ResNet152, Inception_v3) within a Siamese Network Architecture that used triplet loss functions for the identification and re-identification of Sika deer. Subsequently, we then determined the best-performing model based on its ability to capture discriminative features. From this model, we extracted embeddings representing the learned features. We then applied a Support Vector Machine (SVM) to these embeddings to classify individual Sika deer. We analyzed 5169 image datasets consisting of images of seven individual Sika deers captured with three camera traps deployed on farmland in Hokkaido, Japan, for over 60 days. During our analysis, ResNet152 performed exceptionally well, achieving a training accuracy of 0.97, and a validation accuracy of 0.96, with mAP scores for the training and validation datasets of 0.97 and 0.96, respectively. We extracted 128 dimensional embeddings of ResNet152 and performed Principal Component Analysis (PCA) for dimensionality reduction. PCA1 and PCA2, which together accounted for over 80% of the variance collectively, were selected for subsequent SVM analysis. Utilizing the Radial Basis Function (RBF) kernel, which yielded a cross-validation score of 0.96, proved to be most suitable for our research. Hyperparameter optimization using the GridSearchCV library resulted in a gamma value of 10 and C value of 0.001. The OneVsRest SVM classifier achieved an impressive overall accuracy of 0.97 and 0.96, respectively, for the training and validation datasets. This study presents a precise model for identifying individual Sika deer using images and video frames, which can be replicated for other species with unique patterns, thereby assisting conservationists and researchers in effectively monitoring and protecting the species.

**Keywords:** sika deer; siamese network; re-identification; embeddings; PCA; SVM

check for updates

**Citation:** Sharma, S.; Timilsina, S.; Gautam, B.P.; Watanabe, S.; Kondo, S.; Sato, K. Enhancing Sika Deer Identification: Integrating CNN-Based Siamese Networks with SVM Classification. *Electronics* **2024**, *13*, 2067. https://doi.org/10.3390/electronics13112067

Academic Editor: Manohar Das

Received: 22 April 2024
Revised: 13 May 2024
Accepted: 24 May 2024
Published: 26 May 2024

**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

Individual identification lies at the core of conservation efforts and effective management practices. Various traditional methods such as tagging, tattooing, and implants are often used by ecologists for this purpose [1]. However, these approaches often have physiological implications and can adversely affect animal behavior and reproduction [2,3]. Alternatively, camera traps offer a non-invasive solution. Nonetheless, the high volume of images captured annually by camera traps can overwhelm experts, making manual sorting laborious and time consuming. To address these challenges, computer vision techniques

are increasingly employed. These methods streamline species identification from images and facilitate the recognition of unique individual patterns within populations [4,5].

The utilization of Deep Neural Networks (DNNs) in computer vision applications has attracted significant attention within the ecological community, primarily due to their capacity of feature generation and extraction from images captured by camera traps [6]. Within computer vision, two widely employed approaches are classification and similarity learning, which aid in the identification of wildlife [7]. Yet, while significant research has focused on Endangered species such as the Bengal tiger, lion, and zebra listed on the IUCN Red List of Threatened Species [7–9], there has been a notable oversight concerning least-concern species like the Sika deer. This neglect is concerning as it contributes to both economic and biodiversity losses due to the unchecked growth of its population [10].

Sika deer, once abundant in Japan's lowlands by the late 19th century, have since migrated toward mountainous regions with limited human presence, including national parks, since the late 1970s [11]. Our research, conducted in Hokkaido, the northernmost mountainous area of Japan, highlights this migration trend, wherein the burgeoning Sika deer population has become a pressing concern [12]. This population surge can be attributed to the extinction of natural predators like *Canis lupus hodophilax* since 1905 [13]. Presently, the Hokkaido government has also taken measures to protect Sika deer populations [14,15]. Sika deer, being large herbivores and grazers, tend to congregate in herds. However, their overpopulation has led to significant biodiversity loss and agricultural damage, resulting in economic hardship for locals. Moreover, their presence negatively impacts native species and poses challenges to conservation management efforts [16]. Hence, there is an urgent need for comprehensive monitoring and inventory programs to effectively manage conservation [17]. This entails the accurate identification of individual Sika deer. Our study, which focused on this aspect, aimed to assist both conservationists and researchers in addressing these challenges.

Since 2014, deep learning has revolutionized animal identification, with an ever-increasing number of methods being used for animal re-identification [18]. Many of these methods specialize in the identification of specific species with distinct visual features, such as salamanders [19], manta rays [20], cows [21], or Amur tigers [22]. There are multiple deep learning architectures that can be used for these purposes; in particular, Freytag et al. [23] demonstrated the superior performance of the AlexNET CNN architecture over previous methods for identifying two groups of chimpanzees. Building on this success, Brust et al. [24] effectively applied AlexNET to identify gorillas. In a different vein, Schneider et al. [25] demonstrated the efficacy of similarity learning networks for cross-species animal re-identification, eliminating the need for manual feature extraction. This approach yielded promising results on five different species: humans, chimpanzees, whales, fruit flies, and tigers. Some researchers have integrated animal identification methods into frameworks designed to automatically collect information about individual behaviors. Marks et al. [26] developed an end-to-end pipeline capable of extracting specific behaviors (e.g., social grooming or object interaction) as well as animal poses. Their species-agnostic approach was tested on primates and mice with promising results. Similarly, identification has been incorporated into an end-to-end framework focused on fast inference times to analyze the trajectories of individual polar bears over long periods of time [27]. However, there remains a gap in knowledge regarding the use of CNN-based architectures for the individual identification of Sika deer. Therefore, this study was conducted to individually identify Sika deer based on their unique spot shapes, sizes, and patterns.

Concurrently, the process of identifying Sika deer plays a crucial role in regular monitoring efforts. Nonetheless, this task encounters notable challenges due to subtle variations in the size and shape of their distinctive spots, hindering accurate identification. Nevertheless, our automated computer vision approach aids significantly in identifying individual Sika deer (Figure 1), thereby enhancing insights and facilitating practical solutions for population management. The main goal of our research was to automate the identification

of individual Sika deer while simultaneously extracting features for classification, with a focus on regular monitoring to bolster effective conservation practices.

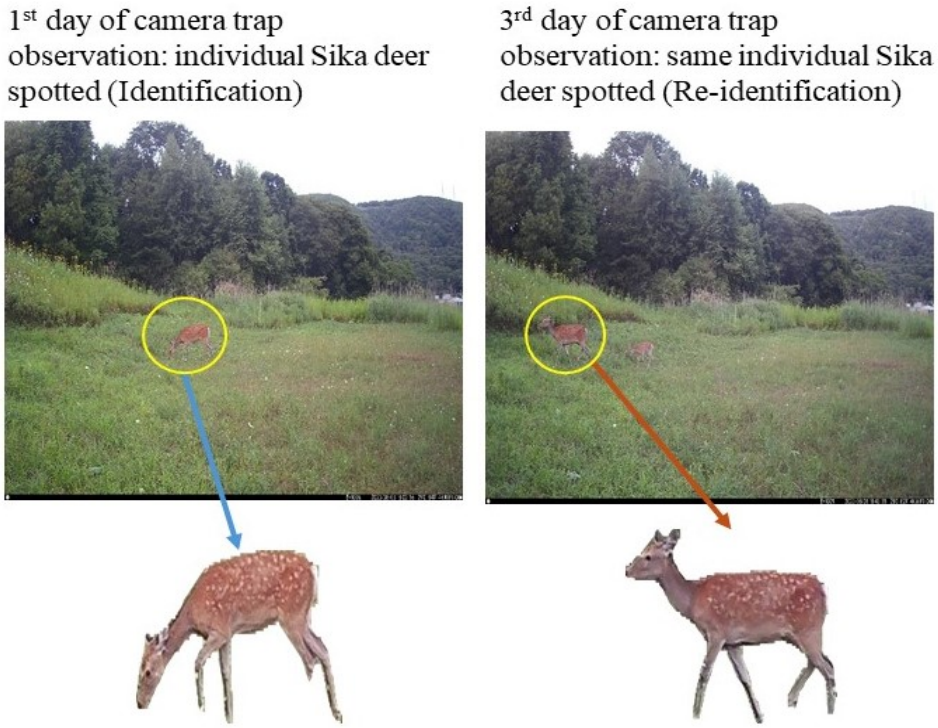


**Figure 1.** Identification and re-identification of individual Sika deer spotted at two different days of camera trap deployment.

## 2. Materials and Methods

Our research encompassed several stages aimed at re-identifying individual Sika deer in their natural habitat. Initially, we collected images and videos from three camera traps, filtering out images devoid of Sika deer presence and manually discarding those with other animals or empty scenes. Video captured was converted into frames, and only those containing Sika deer were retained manually. The compiled Sika deer images and video frames were then preprocessed, including the extraction of frames capturing deer movement and their changing positions relative to the camera traps. Additionally, we cropped individual Sika deer from both images and video frames, organizing them into respective folders corresponding to the seven distinct Sika deer observed. In the subsequent step, the cropped images underwent processing through a Siamese Network utilizing four distinct base models (EfficientNetB7, VGG19, ResNet152, Inception_v3). Among these models, only those exhibiting superior performance metrics were retained for feature extraction from the embedding layer, constituting a deer learning phase. Finally, through the extracted feature embeddings, we employed a Support Vector Machine (SVM) classifier at the machine learning stage to perform the multi-class classification task on individual Sika deer (Figure 2).

Three camera traps were set up in the grasslands near the periphery of the Muroran Institute of Technology (42°22′55″ N–141°02′1″ E) to monitor Sika deer. We selected sites frequented by Sika deer in consultation with university staff members. We chose grasslands as the location for deploying the camera traps due to the preference of Sika deer for open habitats, particularly grasslands, as their foraging sites [28]. The cameras used were Solar Powered 4k-trail cameras manufactured by xiuruidajp (China), H-801A-Pro manufactured by Xiaozhoukeji (China), and HC-801A manufactured by Xiaozhoukeji (China), placed at a height of 100 cm from the base of the tree, at an effective height for Sika deer monitoring [29]. The deployment period spanned 60 days, from 1 June 2023 to 3 August 2023.

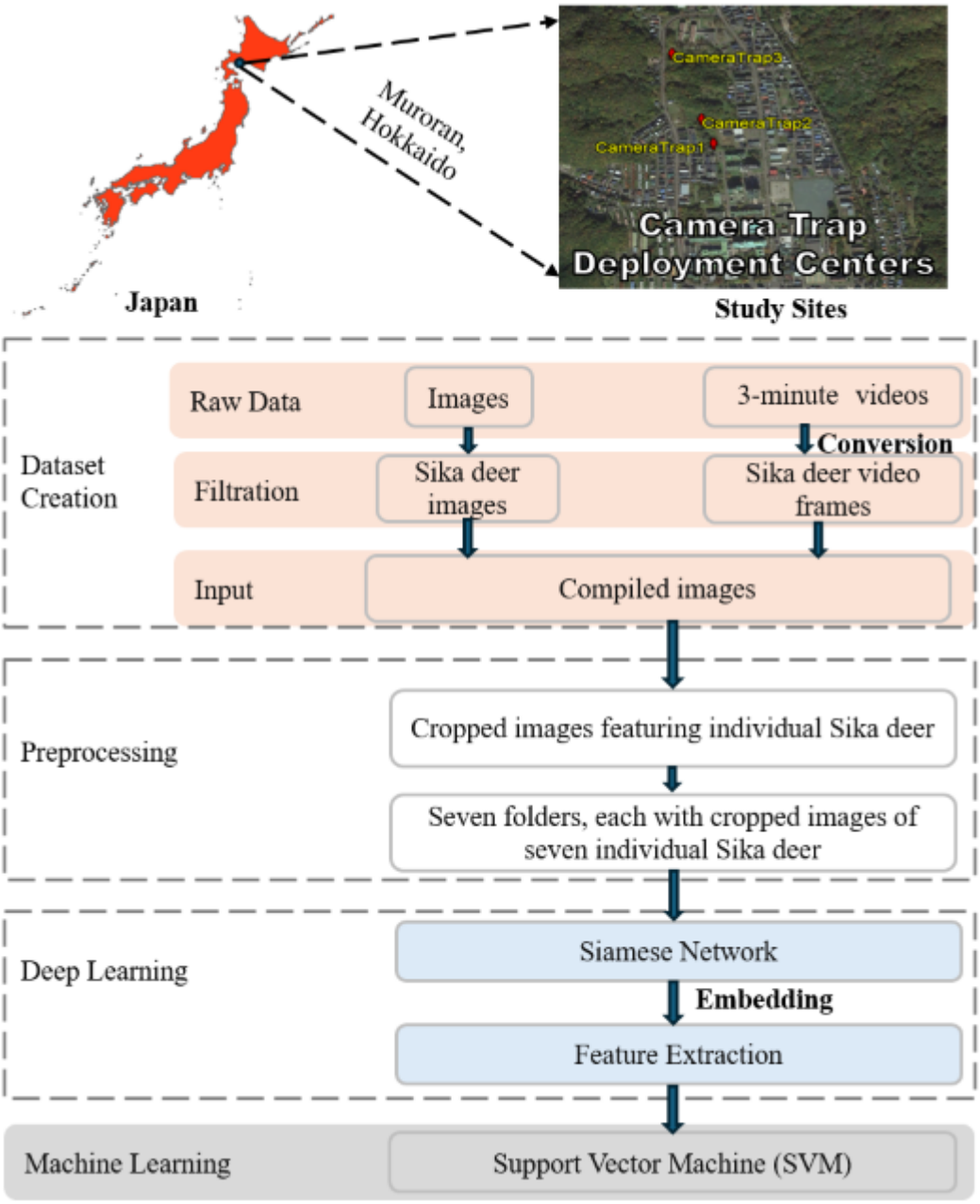


**Figure 2.** Flowchart illustrating the re-identification and classification of individual Sika deer.

### *2.1. Dataset Creations and Preprocessing*

Throughout the deployment period of the camera traps, a combined total of 85 videos and 320 images were documented, highlighting the presence of Sika deer. Video recordings varied from day to day, with some days yielding multiple recordings while others yielded none. An analysis of these recordings and images revealed the presence of seven distinct Sika deer individuals (Figure 3). To train our model, we employed 50 randomly selected videos, alongside all available captured images featuring seven individual Sika deer. The remaining videos were reserved for testing purposes. Each of the 50 selected videos, spanning three minutes, was processed to generate frames at a rate of one frame per second. We manually selected frames from these videos that exhibited Sika deer in motion, with varying orientations and distances from the camera trap's center, ensuring dataset diversity. Recognizing the social behavior of Sika deer, often observed in groups, we cropped individual deer into separate classes for detailed analysis. In total, our dataset comprised

5169 cropped images sourced from the 50 videos and all captured images, encompassing representations of all seven identified Sika deer individuals.

Each individual Sika deer possesses distinctive spot patterns. These patterns, shapes, and sizes of spots present on their bodies are used to assign a unique class ID to each Sika deer. The assignment begins with ID1 for the first individual, ID2 for the second, and so forth, up to ID7 for the seventh individual Sika deer. This classification method enabled us to track and distinguish between multiple Sika deer captured with the camera traps, thereby facilitating the identification of individual Sika deer (Figure 3).

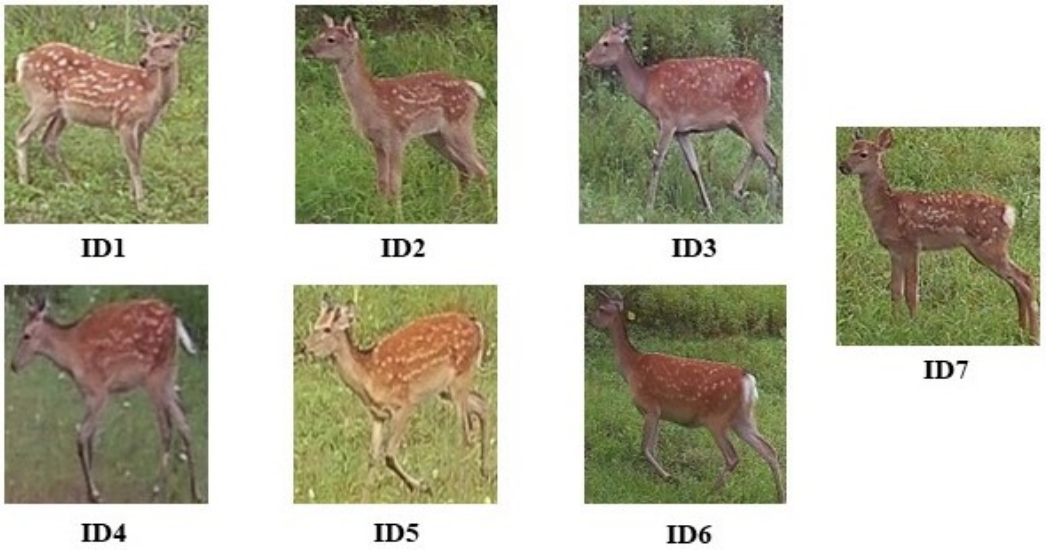


**Figure 3.** Seven individual Sika deer spotted in our deployed camera traps. Each deer was provided with a unique class ID, starting with ID1 for the first individual, ID2 for the second, and continuing up to ID7 for the seventh individual.

### *2.2. Deep Learning Architectures*

In our study, we utilized a Siamese Network due to its widespread acceptance as an architecture for re-identification tasks, particularly renowned for person re-identification [30]. This neural network structure comprises two branches, each constructed with the same Convolutional Neural Network (CNN) backbone, sharing identical weight and bias parameters [20]. During the training phase, image pairs are input into the network, enabling it to learn features from the images. This learning process enables the network to group similar images together while distinguishing dissimilar ones based on the extracted features [31]. In our research, we also utilized similar a Siamese Network Architecture. However, we employed triplet loss function, enhancing the network's performance for Sika deer re-identification. This variant comprised three branches of CNN architectures, providing triplets of images to the network for training. This approach helped the network to learn more nuanced representations and effectively discriminate between different individuals of Sika deer [7].

#### 2.2.1. Siamese Dataset Generation

Initially, we generated triplets using seven distinct classes of images, consisting of anchors, positives, and negatives for the Siamese Network. Each input image was designated as an anchor image, which served as the baseline. Images related to the same class of Sika deer were classified as positive images, while those from different classes of Sika deer, in comparison to the anchor images, were categorized as negative images. We proceeded by dividing the triplets into training (70%) and validation (30%) sets before inputting them into the Siamese Network. Subsequently, we then resized the training and validation triplet datasets to 128 $\times$ 128 pixels. To enrich the training data, we applied several augmentation techniques to the training triplets: rotation of up to 10 degrees, horizontal and vertical

shifting of up to 5%, horizontal flipping, and zooming of up to 20%. Augmentation was carried out to increase the diversity of the training triplet datasets.

#### 2.2.2. Base Models

In our study, we employed four distinct CNN backbones (EfficientNetB7, VGG19, ResNet152, Inception_v3) within the Siamese Network framework. Each of these backbones possesses distinctive characteristics. We employed all four base models to assess which top model architecture performs optimally for the Sika deer re-identification task.

VGG19, a simple and straightforward architecture that achieved second place in the ImageNet competition in 2014, consists of 19 convolutional layers with ReLU activation functions and a $3 \times 3$ kernel size [32]. ResNet152, known for its effectiveness in identification tasks, uses residual connections to address the vanishing gradient problem, boasting 152 fully connected layers [33]. EfficientNetB7 utilizes a compound scaling method to simultaneously adjust the network width, depth, and resolution, optimizing performance and efficiency. It heavily employs depth-wise separable convolutions to reduce parameters and computational cost [34]. Inception_v3 incorporates factorized convolutions to reduce parameters and computational cost and auxiliary classifiers during training to mitigate the vanishing gradient problem and regularize the network [35].

We implemented a fine-tuning strategy for these base models, whereby we designated 25 layers from the top as trainable while the remaining layers were frozen. Following the configuration of the base model, we designed an additional neural network architecture comprising four fully connected layers with dimensions of 512, 256, 128, and 128, respectively. Moreover, we incorporated batch normalization and dropout with an alpha value of 0.3 along with ReLU (Rectified Linear Unit) activation functions between each fully connected layer.

#### 2.2.3. Siamese Network Architecture

Initially, we fed the generated triplets (anchor, positive, and negative images) through three branches of our Siamese Network Architecture, each employed with the same backbone as the base model. This crucial step involved passing the anchor, positive, and negative images independently through the chosen CNN backbone, which served as a feature extractor. After extracting features, the Siamese Network proceeded to compute the triplet loss [36], which is essential for tasks like re-identification during training (Figure 4). Mathematically,

$$\text{Triplet loss}(a, p, n) = \max\{0, d(a, p) - d(a, n) + m\} \quad (1)$$

Here, the variables are denoted as follows:

$a$ = embedding of the anchor images. Anchor images are the reference images through which the model learns to embed similar images closer and dissimilar images farther apart in the embedding space.

$p$ = embedding of the positive images. Positive images are those images that belonged to the same class as the anchor image.

$n$ = embedding of the negative images. Negative images are those that are dissimilar or belong to different classes compared to the anchor image.

$d$ = euclidean distance. It is often used as a metric to measure the dissimilarity or similarity between data points, such as calculating distances between the embedding of two images either anchor-positive and anchor-negative images [26], which were computed using

$$\text{Euclidean distance} = \|f(a) - f(p/n)\|_2 \quad (2)$$

In the above equation, $f(a)$, $f(p)$, and $f(n)$ represent the feature embedding of the anchor, positive, and negative images, respectively.

The positive distance ascertains the similarity between the anchor and positive embedding, while the negative distance measures the dissimilarity between the anchor and

negative embedding. By minimizing the positive distance and maximizing the negative distance, the network ensures that similar images are clustered together in the embedding space, while dissimilar ones are pushed farther apart.

$m$ = margin, which controls the separation between similar and dissimilar vectors in the embedding space. Initially, we set a narrow margin of 0.05. Following this, we iteratively adjusted the margin, increasing it by increments of 0.05 based on the observed overfitting or underfitting in the loss and accuracy plots. Additionally, we evaluated the performance metrics using the Siamese Network Architecture. After careful examination, we determined that a margin value of 0.5 was suitable for Inception_v3 and VGG19, while a margin value of 0.4 worked well for EfficientNetB7 and ResNet152. These margin values provided an approximate fit for our datasets.

Throughout training, the Siamese Network iteratively updated its weight and bias parameters using the Adam optimizer with a learning rate of $1 \times 10^{-6}$ and a batch size of 8 to minimize the triplet loss across the entire datasets. We initially set the number of epochs to 50 for all the base models. However, we ultimately utilized only those epochs where the validation loss was found to be minimal. By backpropagating the loss gradient throughout the network, the model refines its embedding and improves its ability to differentiate between images of Sika deer from different classes.

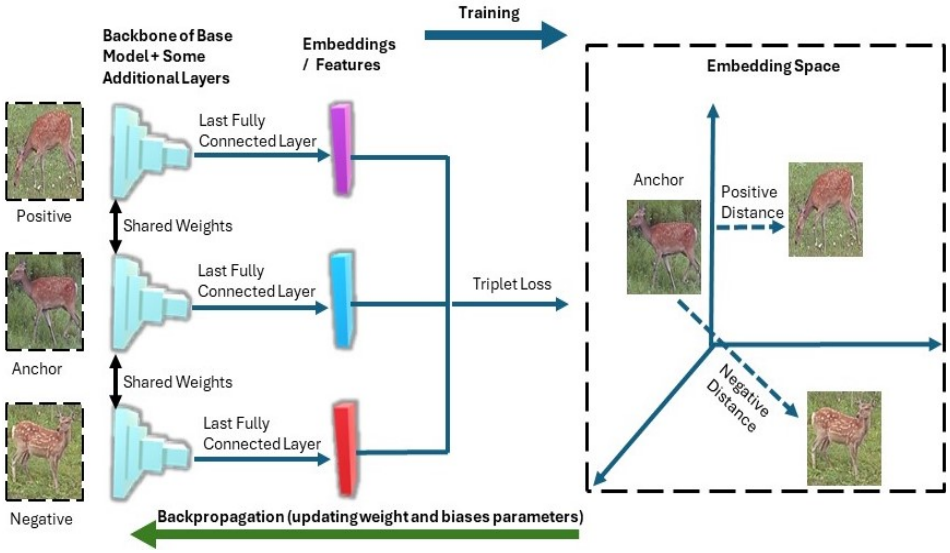


**Figure 4.** Siamese Network Architecture for Sika deer individual identification.

#### 2.2.4. Feature Extraction

We derived embeddings for anchor, positive, and negative image samples from the trained model. These embeddings, in a 128-dimensional vector format, were extracted from the last fully connected layer of the best performance base model [7]. The best-performance model was predicted based on the training and validation accuracy as well as the Mean Average Precision (mAP), which signifies the number of correct matches for the re-identification of Sika deer. As a consequences, three embeddings were generated for each of the anchor, positive, and negative images, effectively capturing the visual traits of the triplet images.

### *2.3. Support Vector Machine*

We generated 128-dimensional embedding vectors from both the training and validation triplets. However, for the classification task, we utilized only the embeddings of the anchor images [7], which served as the baseline images containing all input images. These extracted embeddings underwent Principal Component Analysis (PCA) for dimensional reduction before being input into the Sika deer individual classification task. Notably, PCA1 and PCA2 collectively accounted for over 80% of the variance in both the training

and validation sets (Figure 5). Therefore, we exclusively utilized PCA1 and PCA2 as the subsequent classification machine learning algorithm. In this study, we employed a Support Vector Machine (SVM) for individual Sika deer classification. The SVM is renowned for its capability in classification tasks, enabling the creation of both linear and non-linear decision boundaries. Furthermore, the SVM demonstrates robustness to outliers and gives priority to support vectors for analysis [37].

#### 2.3.1. OneVsRest SVM Classifier

Although the SVM was initially designed for binary classification tasks, our field study revealed the presence of seven distinct individual Sika deer, requiring a multi-class classification strategy. To address this, we utilized the OneVsRest SVM classifier, superficially designed for multi-class classification scenarios. This classification function works similarly to a binary classification model but can accommodate multiple classes [38]. In the OneVsRest SVM classifier, each class of individual Sika deer was trained to be predicted as positive, while the remaining classes were trained as negative.

#### 2.3.2. Kernels

Various kernels are available for SVM classifiers, each with its own characteristics and suitability for different types of datasets. Linear kernels (linear) are commonly used to predict linear combinations of input features, effectively separating datasets with straight lines [39]. Polynomial kernels (poly) are well suited for non-linear datasets, utilizing degree parameters (d) to find optimal values for each dataset [40]. The Radial Basis Function (rbf) kernel operates based on a K-nearest neighborhood, making it ideal for classifying datasets that are not linearly separable, and it is known for its high performance [40]. Lastly, the sigmoid kernel (sigmoid), derived from neural networks, is complex in structure but less widely used due to less-than-ideal backpropagation [39].

Selecting appropriate kernels for the individual Sika deer classification task presented is challenging. Therefore, we computed a 10-fold cross validation score on the training embedding datasets. This involved splitting the training datasets into ten parts (part1, part2, . . . , and part10) to build ten different models to enhance confidence in the performance of our algorithm [41]. The accuracy score obtained from the 10-fold cross-validation provided insight into the suitable kernel types to be used for our datasets (Figure 5).

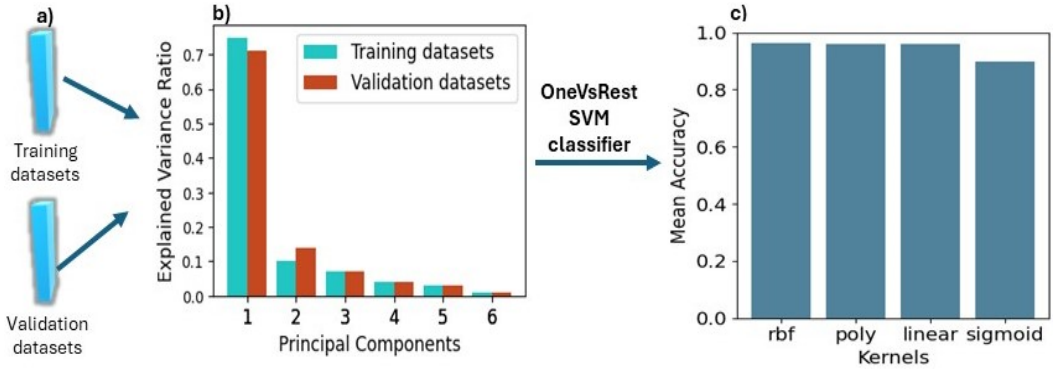


**Figure 5.** Process of selecting kernels for SVM. (**a**) Training and validation anchor emdedding datasets extracted through Siamese Network. (**b**) PCA performed separately on training and validation embedding datasets. (**c**) Selecting appropriate kernels using 10-fold cross validation score using PCA1 and PCA2 of training datasets.

#### 2.3.3. Hyperparameter Tuning

In our datasets, we identified two critical hyperparameters, gamma and C, which demand careful and proper tuning to avoid overfitting or underfitting [40]. Gamma plays a crucial role in shaping the complexity of the decision boundary formed by the SVM model. A lower gamma value yields a smoother decision boundary, potentially causing

underfitting, whereas a higher gamma value results in a more complex decision boundary, possibly leading to overfitting [40]. Similarly, C is a regularization parameter; a smaller C value induces a smoother decision boundary, potentially causing underfitting, while a higher C value generates a more rigid decision boundary, potentially causing overfitting on the datasets [41]. Thus, it is imperative to finely adjust both hyperparameters to suit the training dataset appropriately.

To accomplish this task, we utilized the GridSearch library to determine the optimal gamma and C values [32] within a specified range (−3 to 7, incremented by 3). Subsequently, we retrieved the top five optimal values for gamma and C utilizing the "estimator__gamma" and "estimator__C" attributes provided by the GridSearchCV library. Following the identification of these values, we proceeded to evaluate the performance metrics on both the training and testing datasets to determine whether the model was underfitted, overfitted, or appropriately fitted. This iterative process facilitated the efficient optimization of the hyperparameters, thereby ensuring the robustness of our model.

#### 2.3.4. Performance Metrics

We conducted a thorough analysis of our model's performance using various matrices. The confusion matrices provided insight into true positives (*TP*), which are instances where the model's predicted and observed outcomes were both true; true negatives, (*TN*), which are instances where the model correctly predicted and observed false outcomes; false positives (*FP*), which are instances where the model incorrectly predicted true outcomes; and false negatives (*FN*), which are instances where the model incorrectly predicted false outcomes [42]. These metrics offer a comprehensive assessment of multiclass classification effectiveness, accommodating datasets with both even and uneven distributions [7]. Our evaluation encompassed several key metrics. Accuracy [42], a widely utilized measure, is the ratio of correct predictions to the total number of predictions determined by

$$\frac{TP + TN}{TP + FP + FN + TN} \tag{3}$$

Precision is a critical metric indicating the proportion of true positives among all positives predicted by the model [7], and it is defined as

$$\frac{TP}{TP + FP} \tag{4}$$

Conversely, recall is the proportion of true positives among all actual positive outcomes [7], and it was measured as

$$\frac{TP}{TP + FN} \tag{5}$$

Moreover, we computed the F1-score, which reflects the harmony between precision and recall [43], calculated as follows:

$$2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{6}$$

To provide a comprehensive visualization of the model's performance, we utilized the Area Under the Curve of the Receiver Operating Characteristics (AUC-ROC) curve. This graphical representation illustrates the relationship between TP and FP, offering valuable insights into the model's predictive capabilities [44].

### *2.4. Testing Datasets*

We conducted the testing of our model by randomly selecting five videos per individual Sika deer, ensuring that they were distinct from those used for training and validation. Each chosen video was then converted to frames (one frame per second), and each frame was cropped to isolate the Sika deer. From these cropped frames, we selected 105 instances

of Sika deer (15 per individual) for testing purposes. To facilitate tracking, we renamed each cropped image with the individual Sika deer ID followed by the day as the filename and organized them into a single testing folder. Subsequently, we utilized this testing dataset as input for the best-performing base model backbone of the Siamese Network Architecture, employing anchor images for reference and extracted embeddings for SVM classification. The evaluation of the test dataset was performed using an accuracy matrix, and we visualized the results using a decision boundary analysis.

## 3. Results

We detected seven individual Sika deer from the images and videos recorded with our camera traps. Among the four base models employed in the Siamese Network Architecture, we observed that the ResNet152 model demonstrated quicker convergence with minimal training and validation loss compared to the other models. Specifically, it achieved this in 15 epochs. In contrast, the Inception_v3 backbone required a longer training duration, taking 25 epochs to achieve minimal training and validation loss levels (Figure 6).

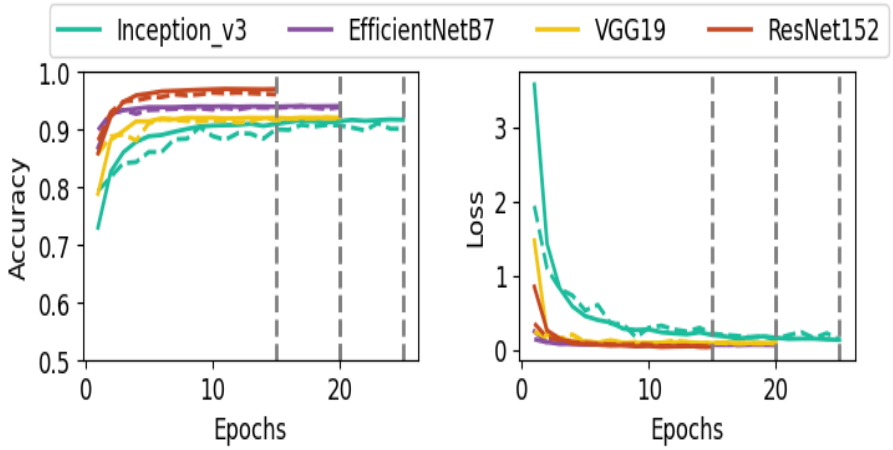


**Figure 6.** The accuracy and loss curves showing the performances of the four base models within the Siamese Network Architecture. Each solid line corresponds to a base model's training datasets, representing both loss and accuracy. Meanwhile, the dotted lines, color-coded accordingly, represent a base model's validation datasets, for loss and accuracy.

We observed that the ResNet152 base model outperformed other base models in terms of convergence, accuracy, and mAP. For ResNet152, the training and validation accuracies were both 0.97 and 0.96, respectively, and the mAP value for both the datasets was 0.96. Conversely, the worst-performing model was Inception_v3, with accuracies of 0.91 for both training and validation datasets, and mAP values of 0.91 and 0.90 for the training and .validation datasets, respectively (Table 1). The Inception_v3 base model demonstrated quicker training times in the Siamese Network Architecture, requiring only 1.28 h on CPU and 0.78 h on GPU. In contrast, VGG19 demanded more training time on both CPU (8.96 h) and GPU (0.92 h). These experiments were conducted using an Intel (R) Core (TM) i7-10700F CPU @ 2.90 GHz, 32.0 GB of RAM, GeForce RTX 3060 Ti, and 8 GB GDDR6 Memory, running Windows 11 Pro, using Tensorflow 2.0 in python version 3.11.7.

**Table 1.** Evaluation of model performance based on convergence epochs, accuracy, mAP, and latency.

| Base Models | Epochs | Training Datasets | | Validation Datasets | | Latency (Hours) | |
|---|---|---|---|---|---|---|---|
| | | Accuracy | mAP | Accuracy | mAP | CPU | GPU |
| Inception_v3 | 25 | 0.91 | 0.91 | 0.91 | 0.90 | 1.28 | 0.78 |
| EfficientNetB7 | 20 | 0.94 | 0.93 | 0.93 | 0.93 | 4.34 | 0.82 |
| VGG19 | 20 | 0.92 | 0.93 | 0.92 | 0.92 | 8.96 | 0.92 |
| ResNet152 | 15 | 0.97 | 0.97 | 0.96 | 0.96 | 3.12 | 0.63 |

In evaluating the performance of our ResNet152 model using the testing datasets, we observed that the model effectively recognized individual Sika deer across different days of camera trap deployment. Here, we present a visual analysis of the model's performance. A sample image, distinguished by a blue rectangle, was used as a reference for comparing the respective Sika deer individuals. Images highlighted with green rectangles signify the accurate identification of the individual Sika deer classes by the model, whereas those within red rectangles denote incorrect class predictions (Figure 7).




**Figure 7.** Samples from testing datasets featuring seven individual Sika deer captured on different days of deployment.

We obtained anchor image embeddings for both the training and validation datasets of ResNet152 for the SVM. Through cross-validation score analysis to ascertain the optimal kernel types for our training embedding datasets, the rbf kernel emerged as the most suitable, boasting a mean accuracy of 0.96. Similarly, through experimentation with different

hyperparameters, like gamma and C, using the GridSearchCV library, we identified a gamma value of 10 and a C value of 0.001 as optimal for our training datasets.

When analyzing the confusion matrix derived from employing PCA1 and PCA2 on both the training and validation datasets with the OneVsRest SVM classifier and utilizing predetermined hyperparameters and rbf kernels, we observed that the classifier effectively distinguished seven distinct Sika deer instances (Figure 8). Notably, it correctly classified ID2, ID6, and ID7 Sika deer. However, there were instances where the classifier misclassified ID1 and ID4 Sika deer (and vice versa), as well as ID3 and ID5 Sika deer (and vice versa), across both the training and validation datasets.

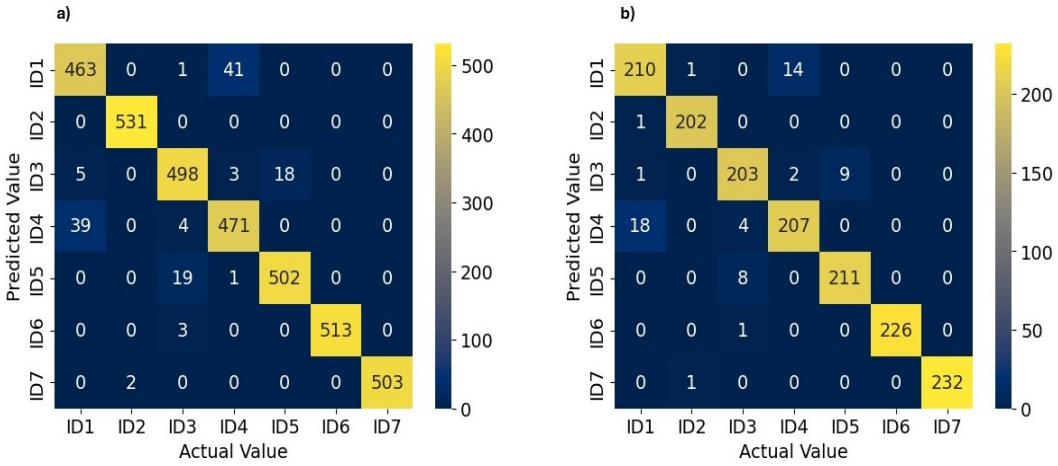


**Figure 8.** Confusion matrices computed from Principal Components. (**a**) Training datasets. (**b**) Validation datasets.

The overall accuracy score for both the training and validation datasets was an impressive 0.96. Specifically, the classifier excelled in accurately classifying ID2, ID6, and ID7, achieving a perfect accuracy of 1 for both the training and validation datasets. For the other individuals, the accuracy remained consistently high, surpassing 0.90 on both datasets. Moreover, other performance metrics such as the precision, recall, and F1-score reflected this trend. ID2, ID6, and ID7 exhibited almost identical scores close to 1 across these metrics on both the training and validation sets. Similarly, for ID1, ID3, ID4, and ID5, these metrics displayed strong performances, also exceeding 0.90 on both datasets as well (Table 2).

**Table 2.** Assessment of the SVM model using various performance metrics.

| Individuals | Training Datasets | | | Validation Datasets | | |
|---|---|---|---|---|---|---|
| | Precision | Recall | F1-Score | Precision | Recall | F1-Score |
| **ID1** | 0.92 | 0.91 | 0.92 | 0.93 | 0.91 | 0.92 |
| **ID2** | 1.00 | 1.00 | 1.00 | 1.00 | 0.99 | 0.99 |
| **ID3** | 0.95 | 0.95 | 0.95 | 0.94 | 0.94 | 0.94 |
| **ID4** | 0.92 | 0.91 | 0.91 | 0.90 | 0.93 | 0.92 |
| **ID5** | 0.96 | 0.97 | 0.96 | 0.96 | 0.96 | 0.96 |
| **ID6** | 0.99 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| **ID7** | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| **Accuracy** | | 0.97 | | | 0.96 | |
| **Macro Averaging** | 0.98 | 0.96 | 0.96 | 0.96 | 0.96 | 0.96 |
| **Weighted Averaging** | 0.99 | 0.96 | 0.96 | 0.96 | 0.96 | 0.96 |

The SVM classifier produced remarkable results, as shown by the AUC-ROC curve (Figure 9). Across both the training and validation datasets, the AUC value exceeded 0.95 for all individual Sika deer, indicating a high degree of accuracy in distinguishing between positive and negative instances of Sika deer individuals. Notably, the classifier accurately classified Sika deer ID2, ID6, and ID7, achieving AUC values close to 1 for each, indicating perfect classification. Additionally, for Sika deer ID1, ID3, ID4, and ID5, the classifier maintained a true positive rate (AUC value) exceeding 0.95, suggesting that the classifier reliably classified these individuals as well, but with a slightly lower level of confidence compared to the previously mentioned Sika deer individuals.

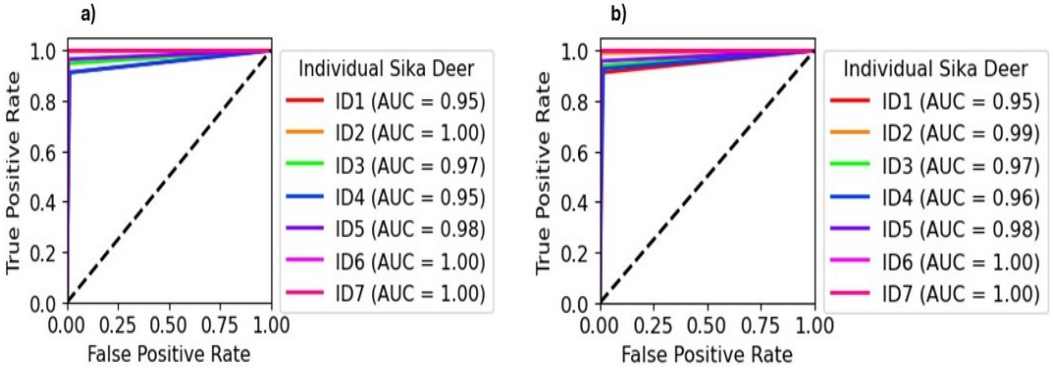


**Figure 9.** AUC-ROC curve of training and validation datasets for seven individual Sika deer. (**a**) Training datasets. (**b**) Validation datasets.

After thorough analysis and optimization, we effectively established the decision boundary utilizing the specified gamma and C hyperparameters across both the training and validation datasets. This approach allowed us to classify individual Sika deer with a significantly increased level of precision and accuracy (Figure 9). Upon closer examination of the decision boundary for individual Sika deer, it becomes evident that those identified as ID2, ID6, and ID7 were classified with notable accuracy by the classifier. However, for individuals ID1, ID3, ID4, and ID5, the classifier exhibited a lower level of confidence in its classification. We found that individual Sika deer positions were consistently located at the same coordinates across both the training and validation datasets, as determined by the decision boundary (Figure 10). Additionally, we conducted tests on the embeddings of testing datasets derived from the Siamese Network, and our findings reveal that individual Sika deer were classified within the decision boundary in the same position coordinates as the training and validation dataset embeddings with a testing accuracy of 0.96 (Figure 10).

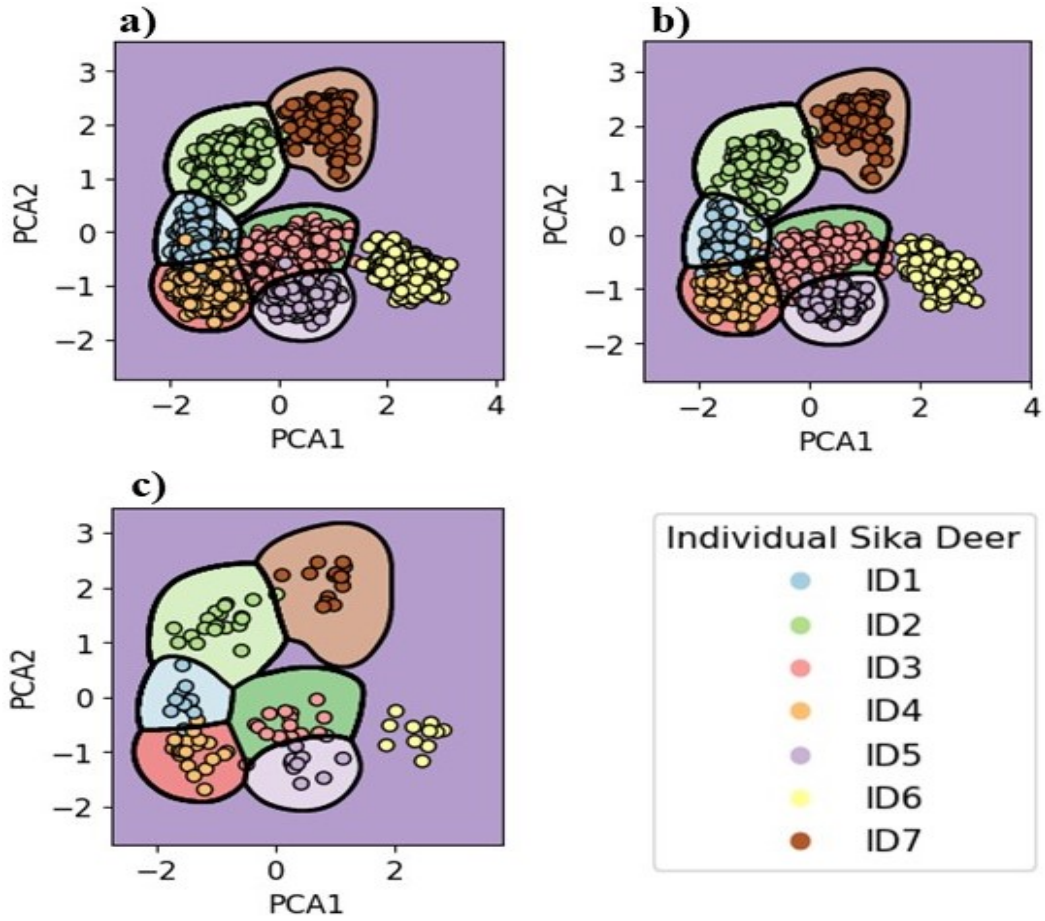


**Figure 10.** Decision boundary of training and validation datasets for classifying seven individual Sika deer. (**a**) Training datasets. (**b**) Validation datasets. (**c**) Testing datasets.

## 4. Discussion

The accurate identification of individual Sika deer is essential for effective management practices and understanding their coexistence with humans. This study focused on automating the re-identification process of individual Sika deer. We utilized the Siamese Network Architecture with triplet loss functions, a method that has seen success in various computer vision applications such as face recognition [45], person re-identification [46], and image retrieval [47]. We applied this architecture to re-identify individual Sika deer captured with our camera traps on different days (Figure 7). Using the Siamese Network Architecture, we extracted features from the embedding layers of the best-performing model. These features were used to form a low-dimensional embedding space utilizing a triplet loss function [48].

For this study, we employed a CNN base model within a Siamese Network framework to identify seven individual Sika deer. Hansen et al. [49] investigated facial recognition using CNN networks on a dataset comprising 10 pig individuals, achieving an accuracy of approximately 97%. Their findings demonstrated the effectiveness of Siamese networks in this context. Uzhinskiy et al. [50] demonstrated the suitability of Siamese networks and the triplet loss function, even in scenarios with limited training data. They showcased impressive results in facial recognition using a dataset consisting of five moss individuals, showing that a Siamese Network with a triplet loss function effectively fit the image data with an accuracy of over 97%. However, despite this, we utilized camera traps over a 60-day period for individual Sika deer identification. This approach mirrors that of Chan et al. [51], who used a similar method for honeybee individual discrimination over a span of just 13 days. Likewise, Ferreira et al. [52] employed bird individual recognition, collecting image datasets over only 15 days.

Among the various base model backbones employed on the Siamese Network Architecture, ResNet152 proved to be highly effective for the individual identification and re-identification of Sika deer. The effectiveness of ResNet152 in identification tasks can be attributed to its residual connections within the architecture pipeline, which effectively address the issue of vanishing gradients [33]. We produced 128-dimensional embedding vectors each from the anchor, positive, and negative images on both the training and validation datasets using the ResNet152 backbone Siamese Network Architecture. However, for the classification task, we utilized only the embedding of anchor images. This decision was made because the triplet loss function employed in this study utilized three identical ResNet152 backbones, each dedicated to the anchor, positive, and negative images, sharing same weight and bias parameters. Additionally, anchor images were chosen as the input images encompassing all the provided images.

The 128-dimensional embedding vectors representing anchor images underwent an initial PCA analysis for dimension reduction, employing a widely recognized technique known for its simplicity and effectiveness [53]. In our study, PCA1 and PCA2 collectively accounted for over 80% of the variance in both our training and validation datasets (Figure 5). Consequently, we focused solely on utilizing these two principal components as input for the classification task.

We highlighted the significance of using the SVM classifier for its ability to establish decision boundaries suitable for both linear and non-linear datasets, while maintaining high precision even with unbalanced data [37]. Originally developed for binary classification, the SVM classifier has been extended to handle multiclass classification tasks. For our study, we employed the OneVsRest SVM classifier, which functions similarly to a binary classification model adapted for multiple classes [38]. Within the OneVsRest framework, each Sika deer was identified as positive for the training, while all other classes were considered negative.

The decision to use an SVM for individual Sika deer classification in our study was deliberate, driven by its versatility in handling both linear and non-linear classification tasks, robustness to outliers, and interpretability. While CNN models are adept at classifying images, the SVM's adaptability to both linear and non-linear patterns makes it ideal for capturing the complex features [54] associated with Sika deer identification. Furthermore, the SVM's robustness to outliers [55] ensures accurate classification even in the presence of irregular data points. In addition, the SVM's ability to define decision boundaries through hyperplanes [55] facilitates the effective separation of classes, contributing to its effectiveness in classification tasks. Furthermore, the SVM has been successfully used in previous studies for similar classification tasks [56,57], further validating our choice.

In our evaluation of kernels for the OneVsRest SVM classifier, the rbf kernel emerged as the most suitable option, achieving a mean accuracy of 0.96 through 10-fold cross-validation. Additionally, the GridSearchCV library optimized the gamma and C values (hyperparameters) exclusively for the OneVsRest SVM classifier with the rbf kernel. We identified optimal settings for a gamma value of 10 and C value of 0.001 for our feature datasets. The uses of 10-fold cross-validation for accuracy assessment and the GridSearch library for hyperparameter tuning are widely recognized techniques for effectively tuning models to datasets [39,58].

Our top-performing model, ResNet152, took 3.12 h when run on a CPU and only 0.63 h on a GPU for feature extraction using the Siamese Network Architecture. Following this, approximately 10.59 s were required on a CPU to conduct Principal Component Analysis (PCA) for the dimension reduction of the extracted features and utilizing PCA1 and PCA2 for SVM classification. VGG19 required more training time compared to other base model backbones for convergence [32], and we observed the same phenomena in our experiment. In contrast, the pre-trained model weights for Inception_v3 were smaller than those of other base models, resulting in very fast convergence training times, which aligns with the findings of Szegedy et al. [35]. Furthermore, we noticed that utilizing the GPU was more computationally efficient than utilizing the CPU.

The misidentification of certain classes of Sika deer, for instances ID1, ID4, ID3, and ID5, might be attributed to minimal differences between them. To explore this further, we initially determined the mean PCA1 and PCA2 values for each class, which served as the centroid. Subsequently, we then evaluated the variation between classes relative to these centroids. This involved calculating the Euclidean distance to quantify the interclass differences between two distinct classes of Sika deer, as defined by the formula

$$\sqrt{((x_2 - x_1)^2 + (y_2 - y_1)^2)} \tag{7}$$

In this context, $x_1$ = the centroid value of PCA1 for one class of Sika deer, while $x_2$ = the centroid value of another class. Similarly, $y_1$ and $y_2$ = the centroid values of PCA2 for each respective class of Sika deer (Figure 11; Table 3). Moreover, to understand the variability within the same class of Sika deer, we computed the intraclass variation. We also used the same euclidean distance Equation (7), where $x_2$ and $x_1$ = the maximum and minimum values of PCA1 within a specific class, while $y_2$ and $y_1$ = the maximum and minimum values of PCA2 within the same class of Sika deer (Figure 11; Table 3). The interclass variation between ID1 and ID4 was found to be very minimal, similar to the situation observed between ID3 and ID5 (Table 3). These minimal interclass variations explicitly demonstrate the misidentification of the respective Sika deer class.

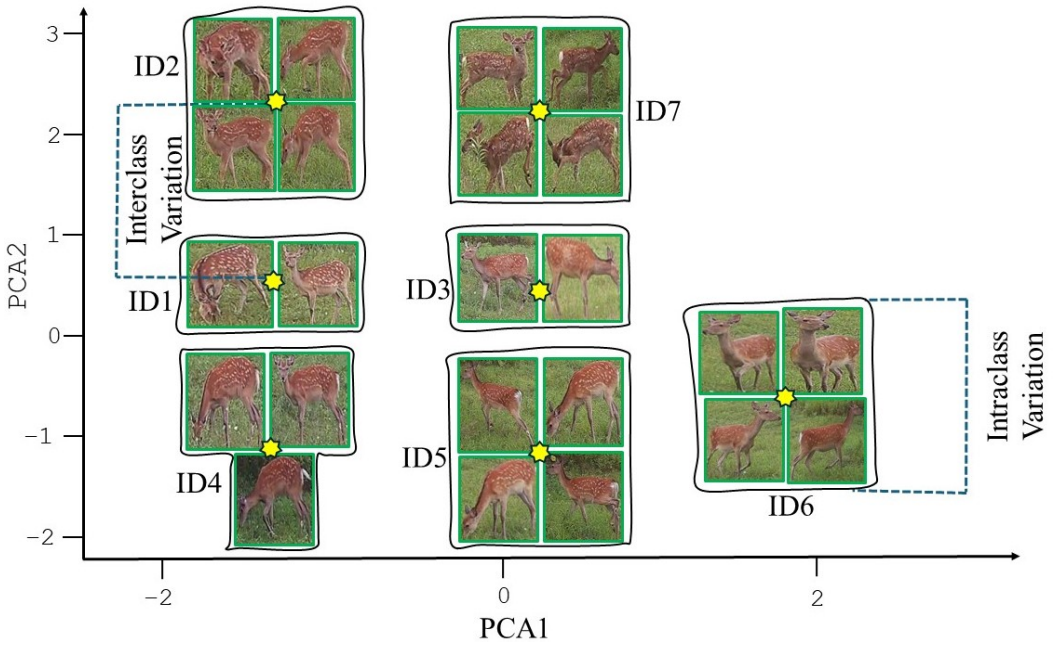


**Figure 11.** Interclass variation in some sample of testing datasets of seven individual Sika deer. Here, yellow stars represent the centroid of each individual Sika deer.

**Table 3.** Interclass and intraclass variations in seven individual Sika deer for testing datasets.

| Individuals | Intraclass Variation | Individuals | Interclass Variation |
|---|---|---|---|
| ID1 | 4.02 | ID1/ID2 | 2.73 |
| ID2 | 5.38 | ID1/ID4 | 0.63 |
| ID3 | 4.74 | ID3/ID5 | 0.81 |
| ID4 | 4.97 | ID3/ID4 | 1.75 |
| ID5 | 2.99 | ID5/ID4 | 1.86 |
| ID6 | 3.03 | ID5/ID6 | 2.28 |
| ID7 | 3.63 | ID2/ID7 | 2.45 |

This research represents a significant advancement in individual Sika deer identification through integrating deep learning and machine learning approaches. Nevertheless, it is imperative to acknowledge certain limitations. Initially, our research was carried out on seven individual Sika deer that were concurrently captured with our camera traps, thereby limiting the model's effectiveness in analyzing a broader population. We propose to expand this study to encompass a broader spectrum of Sika deer population. Secondly, we suggest exploring various other loss functions, such as contrastive loss, in addition to the triplet loss function employed in this study, to further elucidate the generalization of the Siamese Network on a deeper level. Furthermore, we propose to extend this approach to the individual identification of terrestrial and aquatic animals and plants due to its feasibility of implementation. This study utilized Sika deer videos and images captured only from three camera traps installed over a period of 60 days during the summer. However, we recommend deploying multiple camera traps over extended periods to document a wider range of environmental conditions. While our methods could be the basis for similar studies elsewhere, we recognize the need for additional research in different environments to thoroughly verify and potentially adopt our approach for Sika deer individual identification. Additionally, we propose replicating this study using images obtained using Unmanned Aerial Vehicles (UAVs), which provides a top-down view of individual Sika deer, a crucial aspect for re-identification.

## 5. Conclusions

In summary, our study utilized four base models (EfficientNetB7, VGG19, ResNet152, Inception_v3) within Siamese Network Architectures, employing the anchor, positive, and negative images for Sika deer identification. Among these models, ResNet152 emerged as the top performer. We extracted 128-dimensional vector embeddings from the ResNet152 backbone for both training and validation datasets post training. These embeddings were subsequently fed into an SVM for the classification of seven individual Sika deer, achieving training and validation accuracies of 0.97 and 0.96, respectively. The SVM employed an rbf kernel with a mean accuracy score of 0.96, along with hyperparameters gamma = 10 and C = 0.001, facilitated through a OneVsRest SVM classifier. Our study's main contribution is in successfully integrating state-of-the-art base models into Siamese Network Architectures for the identification of Sika deer, particularly highlighting the effectiveness of ResNet152 in this regard. This innovative method not only showcases a superior accuracy but also establishes a framework for efficiently and accurately identifying individual Sika deer in wildlife conservation endeavors. Additionally, our research underscores the significance of employing machine learning techniques, particularly SVMs in ecological studies, emphasizing their role in enhancing wildlife monitoring and management practices, especially in the context of Sika deer conservation.

**Author Contributions:** This study was designed by S.S., K.S. and B.P.G., S.S. conducted the collection and analysis of field datasets. The first draft of the manuscript was written by S.S. Following this, S.S., S.T., B.P.G., S.W., S.K. and K.S. contributed to the manuscript's review and editing process, ultimately granting final approval. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was supported by JSPS KAKENHI (#Grant Number: JP23K23728).

**Data Availability Statement:** Data analysis code and sample raw data can be accessed publicly on github: https://github.com/sand198/Sika_deer_Siamese_reidentification.git (accessed on 21 April 2024). Additionally, instructions on executing the code are also provided on the platform.

**Acknowledgments:** We extend our appreciation for the assistance supported through the Project Based Learning-AI (PBL-AI) program and the Ministry of Education, Culture, Sports, Science, and Technology (Monbukagakusho: MEXT) scholarship awarded to Sandhya Sharma and Suresh Timilsina. We are deeply grateful to Rakshya Poudel and Dipak Khatri for their invaluable assistance in data collection in the field.

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

## References


1. Lim, M.A.; Defensor, E.B.; Mechanic, J.A.; Shah, P.P.; Jaime, E.A.; Roberts, C.R.; Hutto, D.L.; Schaevitz, L.R. Retrospective analysis of the effects of identification procedures and cage changing by using data from automated, continuous monitoring. *J. Am. Assoc. Lab. Anim. Sci. (JAALAS)* **2019**, *58*, 126–141. [CrossRef] [PubMed]
2. Schacter, C.R.; Jones, I.L. Effects of geolocation tracking devices on behavior, reproductive success, and return rate of aethia auklets: An evaluation of tag mass guidelines. *Wilson J. Ornithol.* **2017**, *129*, 459–468. [CrossRef]
3. Wright, D.W.; Stein, L.H.; Dempster, T.; Oppedal, F. Differential effects of internal tagging depending on depth treatment in atlantic salmon: A cautionary tale for aquatic animal tag use. *Curr. Zool.* **2019**, *65*, 665–673. [CrossRef] [PubMed]
4. Nguyen, H.; Maclagan, S.J.; Nguyen, T.D.; Nguyen, T.; Flemons, P.; Andrews, K.; Ritchie, E.G.; Phung, D. Animal recognition and identification with deep convolutional neural networks for automated wildlife monitoring. In Proceedings of the IEEE International Conference on Data Science and Advanced Analytics, Tokyo, Japan, 19–21 October 2017; pp. 327–338.
5. Verma, G.K.; Gupta, P. Wild animal detection using deep convolutional neural network. In Proceedings of the 2nd International Conference on Computer Vision and Image Processing, Las Vegas, NV, USA, 27–29 August 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 327–338.
6. Nepovinnykh, E.; Eerola, T.; Biard, V.; Mutka, P.; Niemi, M.; Kunnasranta, M.; Kalviainen, H. SealID: Saimaa Ringed Seal Re-Identification Dataset. *Sensors* **2022**, *22*, 7602. [CrossRef] [PubMed]
7. Dlamini, N.; van Zyl, T.L. Automated identification of individuals in wildlife population using Siamese neural networks. In Proceedings of the IEEE 7th International Conference on Soft Computing and Machine Intelligence, Stockholm, Sweden, 14–15 November 2020; pp. 224–228.
8. Shukla, A.; Singh, C.G.; Gao, P.; Onda, S.; Anshumaan, D.; Anand, S.; Farrell, R. A hybrid approach to tiger re-identification. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, Seoul, Republic of Korea, 27–28 October 2019.
9. Das, A.; Sinha, S.S.S.; Chandru, S. Identification of a Zebra Based on Its Stripes through Pattern Recognition. In Proceedings of the International Conference on Design Innovations for 3Cs Compute Communicate Control (ICDI3C), Banglore, India, 10–11 June 2021; pp. 120–122.
10. Dhakal, T.; Kim, T.; Kim, S.; Tiwari, S.; Kim, J.; Jang, G.; Lee, D. Distribution of Sika deer (*Cervus nippon*) and the bioclimatic impact on their habitats in South Korea. *Sci. Rep.* **2023**, *13*, 19040. [CrossRef] [PubMed]
11. McCullough, D.R.; Takatsuki, S.; Kaji, K. *Sika Deer: Biology and Management of Native and Introduced Populations*; Springer: Berlin/Heidelberg, Germany, 2008.
12. Nabata, D.; Masuda, R.; Takahashi, O. Bottleneck effects on the Sika deer Cervus nippon population in Hokkaido, revealed by ancient DNA analysis. *Zool. Sci.* **2004**, *21*, 473–481. [CrossRef] [PubMed]
13. Nakajima, N. Changes in distributions of wildlife in Japan. In *Rebellion of Wildlife and Collapse of Forest*; Forest and Environment Research Association, Japan, Ed.; Shinrinbunka Association: Tokyo, Japan, 2007; pp. 57–68.
14. Hokkaido Government. *Result of a Survey Related to Sika Deer and Brown Bear Sightings in Hokkaido*; Hokkaido Nature Preservation Division: Sapporo, Japan, 1986. (In Japanese)
15. Hokkaido Government. *Result of a Survey Related to Sika Deer and Brown Bear Sightings in Hokkaido*; Hokkaido Institute of Environmental Sciences: Sapporo, Japan, 1994. (In Japanese)
16. McMillan, S.E.; Dingle, C.; Allcock, J.A.; Bonebrake, T.C. Exotic animal cafes are increasingly home to threatened biodiversity. *Conserv. Lett.* **2021**, *14*, e12760. [CrossRef]
17. Kalb, D.M.; Delaney, D.A.; DeYoung, R.W.; Bowman, J.L. Genetic diversity and demographic history of introduced sika deer on the Delmarva Peninsula. *Ecol. Evol.* **2019**, *9*, 11504–11517. [CrossRef] [PubMed]
18. Schneider, S.; Tayor, G.W.; Linquist, S.; Kremer, S.C. Past, present and future approaches using computer vision for animal re-identification from camera trap data. *Methods Ecol. Evol.* **2019**, *10*, 461–470. [CrossRef]
19. Matthé, M.; Sannolo, M.; Winiarski, K.; Spitzen van der Sluijs, A.; Goedbloed, D.; Steinfartz, S.; Stachow, U. Comparison of photo-matching algorithms commonly used for photographic capture–recapture studies. *Ecol. Evol.* **2017**, *7*, 5861–5872. [CrossRef]
20. Moskvyak, O.; Maire, F.; Dayoub, F.; Armstrong, A.O.; Baktashmotlagh, M. Robust re-identification of manta rays from natural markings by learning pose invariant embeddings. In Proceedings of the 2021 Digital Image Computing: Techniques and Applications (DICTA), Gold Coast, Australia, 29 November–1 December 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 1–8.
21. Bergamini, L.; Porrello, A.; Dondona, A.C.; Del Negro, E.; Mattioli, M.; D'alterio, N.; Calderara, S. Multi-views embedding for cattle re-identification. In Proceedings of the 2018 14th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS), Canaria, Spain, 26–29 November 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 184–191.
22. Shi, C.; Liu, D.; Cui, Y.; Xie, J.; Roberts, N.J.; Jiang, G. Amur tiger stripes: Individual identification based on deep convolutional neural network. *Integr. Zool.* **2020**, *15*, 461–470. [CrossRef] [PubMed]
23. Freytag, A.; Rodner, E.; Simon, M.; Loos, A.; Kühl, H.S.; Denzler, J. Chimpanzee faces in the wild: Log-euclidean CNNs for predicting identities and attributes of primates. In Proceedings of the German Conference on Pattern Recognition, Hannover, Germany, 12–15 September 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 51–63.
24. Brust, C.A.; Burghardt, T.; Groenenberg, M.; Kading, C.; Kuhl, H.S.; Manguette, M.L.; Denzler, J. Towards automated visual monitoring of individual gorillas in the wild. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Venice, Italy, 22–29 October 2017; pp. 2820–2830.

25. Schneider, S.; Taylor, G.W.; Kremer, S.C. Similarity Learning Networks for Animal Individual Re-Identification—Beyond the Capabilities of a Human Observer. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV) Workshops, Snowmass Village, CO, USA, 1–5 March 2020.
26. Marks, M.; Jin, Q.; Sturman, O.; von Ziegler, L.; Kollmorgen, S.; von der Behrens, W.; Mante, V.; Bohacek, J.; Yanik, M.F. Deep-learning-based identification, tracking, pose estimation and behaviour classification of interacting primates and mice in complex environments. *Nat. Mach. Intell.* **2022**, *4*, 331–340. [CrossRef] [PubMed]
27. Zuerl, M.; Stoll, P.; Brehm, I.; Raab, R.; Zanca, D.; Kabri, S.; Happold, J.; Nille, H.; Prechtel, K.; Wuensch, S.; et al. Automated Video-Based Analysis Framework for Behavior Monitoring of Individual Animals in Zoos Using Deep Learning—A Study on Polar Bears. *Animals* **2022**, *12*, 692. [CrossRef] [PubMed]
28. Takatsuki, S. Edge effects created by clear-cutting on habitat use by sika deer on Mt. Goyo, northern Honshu, Japan. *Ecol. Res.* **1989**, *4*, 287–295. [CrossRef]
29. Ikeda, T.; Takahashi, H.; Yoshida, T.; Igota, H.; Kaji, K. Evaluation of camera trap surveys for estimation of Sika deer herd composition. *MAMM Study* **2013**, *38*, 29–33. [CrossRef]
30. Melekhov, I.; Kannala, J.; Rahtu, E. Siamese network features for image matching. In Proceedings of the 23rd International Conference on Pattern Recognition (ICRR), Cancun, Mexico, 4–8 December 2016; pp. 378–383.
31. Martín-Gómez, L.; Pérez-Marcos, J.; Cordero-Gutiérrez, R.; de la Iglesia, D. Promoting social media dissemination of digital images through CBR-based tag recommendation. *DITTET* **2022**, *5*, 37002. [CrossRef]
32. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. In Proceedings of the 3rd International Conference on Learning Representations (ICLR), San Diego, CA, USA, 7–9 May 2015; pp. 1–14.
33. 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, Boston, MA, USA, 7–12 June 2015; pp. 770–778.
34. Tam, M.; Le, Q.V. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In Proceedings of the 36th International Conference on Machine Learning (ICML), Long Beach, CA, USA, 9–15 June 2019; pp. 10691–10700.
35. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojina, Z. Rethinking the Inception architecture for computer vision. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2818–2826.
36. Wang, J.; Song, Y.; Leung, T.; Rosenberg, C.; Wang, J.; Philbin, J.; Chen B.; Wu, Y. Learning fine-grained image similarity with deep ranking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1386–1393.
37. Sowmya, M.; Balasubramanian, M.M.; Vaidehi, K. Classification of animals using MobileNet with SVM classifier. In *Computational Methods and Data Engineering (ICCMDE)*; Springer Nature: Singapore, 2021; pp. 347–358.
38. Xu, L.; Zhao, G.Z.; Gu, H. Novel one-vs-rest classifier based on SVM and multi-spheres. *J. Zhejiang Univ. Sci.* **2009**, *43*, 303–308.
39. Anonymous. Guide on Support Vector Machine (SVM) Algorithm. Analytics Vidhya. 2024. Available online: https://www.analyticsvidhya.com/blog/2021/10/support-vector-machinessvm-a-complete-guide-for-beginners/ (accessed on 10 March 2024).
40. Anonymous. Support Vector Machine—Simply Explained. Medium. 2019. Available online: https://towardsdatascience.com/support-vector-machine-simply-explained-fee28eba5496 (accessed on 10 March 2024).
41. Anonymous. Cross-Validation: Evaluating Estimator Performance. Scikit Learn. 2024. Available online: https://scikit-learn.org/stable/modules/cross_validation.html (accessed on 21 January 2024).
42. Sharma, S.; Sato, K.; Gautam, B.P. A Methodological Literature Review of Acoustic Wildlife Monitoring Using Artificial Intelligence Tools and Techniques. *Sustainability* **2023**, *15*, 7128. [CrossRef]
43. Xie, J.; Zhu, M.; Hu, K.; Zhang, J.; Hines, H.; Guo, Y. Frog calling activity detection using lightweight CNN with multi-view spectrogram: A case study on Kroombit tinker frog. *Mach. Learn. Appl.* **2021**, *7*, 100202. [CrossRef]
44. Allen, A.N.; Harvey, M.; Harrell, L.; Jansen, A.; Merkens, K.P; Wall, C.C.; Cattiau, J.; Oleson, E.M. A Convolutional Neural Network for Automated Detection of Humpback Whale Song in a Diverse, Long-Term Passive Acoustic Dataset. *Front. Mar. Sci.* **2021**, *8*, 607321. [CrossRef]
45. Schoff, F.; Kalenichenko, D.; Philbin, J. FaceNet: A Unified Embedding for Face Recog nition and Clustering. *arXiv* **2015**, arXiv:1503.03832.
46. Hermans, A.; Beyer, L.; Leibe, B. In defense of the triplet loss for person re-identification. *arXiv* **2017**, arXiv:1703.07737.
47. Gordo, A.; Almazán, J.; Revaud, J.; Larlus, D. Deep image retrieval: Learning global representations for image search. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 241–257.
48. Chou, Y.; Chang, C.; Remedios, S.; Butman, J.A.; Chan, L. Automated classification of resting-state fMRI ICA components using a Deep Siamese Network. *Front. Neurosci.* **2022**, *16*, 768634. [CrossRef]
49. Hansen, M.F.; Smith, M.L.; Smith, L.N.; Salter, M.G.; Boxter, E.M.; Farish, M.; Grieve, B. Towards on farm pig face recognition using Convolutional Neural Networks. *Comput. Ind.* **2018**, *98*, 145–152. [CrossRef]
50. Uzhinskiy, A.V.; Gennadii, A.O.; Pavel, V.G.; Andrei, V.N.; Artem, A.S. One shot learning with triplet loss for vegetation tasks. *Comput. Opt.* **2021**, *45*, 608–614. [CrossRef]

51. Chan, J.; Carrion, H.; Egret, R.M.; Agosto-Rivera, J.L.; Giray, T. Honeybee re-identification in video: New datasets and impact of self-supervision. In Proceedings of the VISGRAPP, Online, 6–8 February 2022; pp. 517–525.
52. Ferreira, A.C.; Silva, L.R.; Renna, F.; Brandi, H.B.; Renoult, J.P.; Farine, D.R.; Covas, R.; Doutrelant, C. Deep learning based methods for individual recognition in small birds. *Methods Ecol. Evol.* **2019**, *11*, 1072–1085. [CrossRef]
53. Velliangiri, S.; Alagumuthukrishnan, S.; Thankumar Joseph, S.I. A review of dimensionality reduction techniques for efficient computation. *Procedia Comput. Sci.* **2019**, *165*, 104–111. [CrossRef]
54. Rana, A.; Vaidya, P.; Gupta, G. A comparative study of quantum support vector machine algorithm for handwritten recognition with support vector machine algorithm. *Mater. Today Proc.* **2022**, *56*, 2025–2030. [CrossRef]
55. Singh, V.; Sharma, N.; Singh, S. A review of imaging techniques for plant diseases detection. *Artific. Intellig. Agricult.* **2020**, *4*, 229–242. [CrossRef]
56. Zhang, L.; Yang, B. Research on recognition of maize diseases based on mobile internet and support vector machine techniques. *Adv. Mater. Res.* **2014**, *905*, 659–662. [CrossRef]
57. Zhang, Z.; He, X.; San, L.; Guo, J.; Wang, F. Image recognition of maize leaf diseases based on GA-SVM. *Chem. Eng. Trans.* **2015**, *46*, 199–204.
58. Anonymous. Tune Hyperparameters with GridSearchCV. Analytics Vidhya. 2023. Available online: https://www.analyticsvidhya.com/blog/2021/06/tune-hyperparameters-with-gridsearchcv/ (accessed on 15 January 2024).


**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.