Optimizing ResNet50 for Non-Invasive Cattle Identification: A Study on Color Block Pattern Recognition

by

Shan Kuang

This dissertation was submitted in part fulfilment of requirements for the degree of

MSc Artificial Intelligence and Applications

Department of Computer and Information Sciences
University of Strathclyde

August 2024

Declaration

This dissertation is submitted in part fulfilment of the requirements for the degree of MSc of the University of Strathclyde. I declare that this dissertation embodies the results of my own work and that it has been composed by myself. Following normal academic conventions, I have made due acknowledgement to the work of others. No ethics approval was required.

I give permission to the University of Strathclyde, Department of Computer and Information Sciences, to provide copies of the dissertation, at cost, to those who may in the future request a copy of the dissertation for private study or research.

I give permission to the University of Strathclyde, Department of Computer and Information Sciences, to place a copy of the dissertation in a publicly available archive.

Yes [☒] No [☐]

I declare that the word count for this dissertation (excluding title page, declaration, abstract,

Acknowledgement, table of contents, list of illustrations, references and appendices) is 9411 words.

I confirm that I wish this to be assessed as a Type V Dissertation.

Signature:

Date: 14th August 2024

Abstract

This dissertation explores an innovative approach to cattle identification using deep learning techniques, specifically focusing on the unique color block patterns on cattle backs. Leveraging the ResNet50 architecture, we developed a model capable of accurately identifying individual cattle, addressing the limitations of traditional identification methods such as ear tags or implants.

Our research utilized *'The Cows2021 Dataset'*, comprising 181 Holstein-Friesian cattle with approximately 8,000 images. We conducted comprehensive experiments comparing deep learning models, image preprocessing techniques, and hyperparameter settings. The optimized ResNet50 model achieved 98.2% test accuracy and 97.3% transfer learning accuracy, demonstrating robust performance in cattle identification.

Key findings include the superiority of ResNet50 over custom models and EfficientNetB0 regarding accuracy and stability. We also investigated the impact of image preprocessing, revealing that while color information is crucial, grayscale images retain sufficient distinguishing features. Conversely, reducing image resolution significantly affected model performance, highlighting the importance of spatial details in cattle identification.

The ability to distinguish cattle with complex or similar patterns remains a challenge despite high accuracy overall. Our error analysis revealed that the model tends to over-rely on local features and struggles with complex edge patterns, indicating areas for further investigation.

This research contributes to the field of livestock management by proposing a non-invasive, scalable, and potentially more animal-friendly method of cattle identification. The findings lay the groundwork for further advancements in automated livestock monitoring and management systems, with potential applications extending to wildlife conservation efforts.

CONTENTS

1 Introduction and Rationale

In recent years of the global agricultural section, we ′ ve witnessed significant improvement in livestock management. Particularly in the field of identification and live tracking of individual animals (Gunaratnam et al., 2024). Among agricultural development, the livestock management is crucial for enhancing animal husbandry. Including improves productivity, health monitoring, and breeding practices. Cattle are one of the most important species among livestock because they produce high quality protein for dairy and beef industry. This dissertation delves into the cutting-edge intersection of deep learning and animal husbandry, exploring a novel approach to cattle identification that promises to enhance both productivity and animal welfare.

Current identification methods require marking cattle or chip injection are inefficient for long-term identification and not ethically friendly. These manual methods for cattle identification are labour-intensive, repetitive, error-prone, and not sustainable in large farms. This highlights the urgent need for automated identification systems that leverage technological advances to improve accuracy and reduce labour costs.

In recent years, various of methods have been explored, from using cow nose to iris (Kumar et al., 2017), with designs in multiple aspects. However, those methods are still requiring high-resolution in lens and high computational power for real-time monitoring. Therefore, we aim to develop a method that use fewer technology resources while maintaining accuracy in cattle identification. This approach will be suitable for not only the large farm, but also those small farms who wish to save labour in enhancing cattle management.

This dissertation is guided by the following research questions:

1. How can self-supervised learning with deep neural network, specifically ResNet-50, being optimized to enhance the accuracy of cattle identification from visual data?

2. What is the impact of different image prepossessing techniques, such as resolution scaling and colour adjustments, affect the accuracy and efficiency of the identification system?

This research aims to:

- Using ResNet-50 to develop a self-supervised learning model for identify the individual cattle based on Cows2021 Dataset with features modification.
- Evaluate the effectiveness of different image prepossessing methods in maintaining the accuracy while reducing the computational power.

Our research leverages the power of deep learning, specifically the ResNet50 architecture, to recognize cattle based on their natural coat patterns. This approach offers several compelling advantages:

Non-invasive identification: By utilizing existing physical features, we eliminate the need for artificial markings or implants, prioritizing animal comfort and welfare.

Scalability: Once trained, the system can potentially identify large herds quickly and accurately, streamlining management processes.

Cost-effectiveness: While initial setup may require investment in imaging technology, the long-term operational costs could be significantly lower than traditional tagging systems.

Data integration potential: A vision-based system opens doors to integrating identification with other visual health and behavior monitoring tools, creating a more comprehensive management ecosystem.

Adaptability: The principles developed here could extend beyond cattle to other livestock and even wildlife conservation efforts, broadening the impact of this research.

By pushing the boundaries of what's possible in animal identification, we're not just solving a technical challenge – we're paving the way for more humane, efficient, and sustainable agricultural practices. This research stands at the forefront of a paradigm shift in livestock management, where cutting-edge technology meets age-old farming wisdom to create a better future for animals and humans alike.

2 Literature Review

2.1 Animal Detection and Livestock Management

2.1.1 Animal Detection

Animal detection and identification have become a crucial part of different fields of science, from Wildlife conservation and bioacoustics to Agricultural management (Senbagam & Bharathi, 2024). This technique aims to accurately detect and identify animals, allowing for better resource management and enhancing agricultural productivity while maintaining animal welfare. Traditional methods of livestock management often rely on first marking the animal with manual observation, which is very time-consuming, labor-intensive and easily disturbed by human error (Kavitha et al., 2023). Recent developments in technology, especially in computer vision and machine learning, have evolved animal detection methods, making them more efficient and reliable in livestock management. Many applications have been developed based on this, such as individual tracking, health monitoring, measuring body size and weight and productivity analysis (Gunaratnam et al., 2024).

2.1.2 Current Livestock Management System

By researching the current livestock management industry, the identification systems are always incorporate various technologies such as ear tags, collars, barcodes, Electronic Identification (EID) stick readers, weigh heads, indicators, load bars, and accessories to enhance accuracy, efficiency, and data reliability (Allflex n.d.). However, those equipment can cause various of damage according to species. For example, the sensor ear tags have been found to cause crust and occasional pain reactions in cattle, some cases might requiring veterinary intervention and leaning to ear injuries or scarring over long-term use (Gobbo, 2023). Similarly, electronic leg and ear tags in goats showed a high readability rate but were associated with foot and udder infections (Kandemir et al., 2023).

2.2 Cattle Identification Features in Computer Vision

Researchers have shown significant interest in reducing the harm to animals caused by livestock management since it not only does they affect animal welfare, they also take a toll on the farm economy. Thus, people are trying to identify and manage them by exploring potential distinguishing features using computer vision.

2.2.1 Biometrics Features

Starting with the premise that unique biometric features function similarly to human fingerprints, the paper *“Analytical Study of Animal Biometrics: A Technical Survey”* comprehensively reviews various biometric methods for cattle identification. These methods include iris patterns, muzzle prints, and retinal vascular patterns, each described with advantages and challenges (Kumar et al., 2017). Much research has further investigated these patterns in developing practical methods for real-world applications.

For example, Diwakar Agarwal (2023) applied a color-based skin detection approach to segment cattle in images captured from mobile cameras, and the Bayes classifier is used for maximum likelihood estimation in HSV color space (Agarwal, 2023).

Moreover, the paper *"Cattle identification from muzzle print image pattern using hybrid feature descriptors and SVM"* explores the combination of digital image processing and machine learning for high-accuracy identification (Kaur et al., 2023). This approach proves the possibility of identifying cattle using muzzle print is workable under hybrid techniques.

Another notable research, *"Evaluation of retinal imaging technology for the biometric identification of bovine animals in Northern Ireland"* evaluates the use of retinal patterns in animal identification, shows the uniqueness and reliability of retinal features as an alternative to traditional methods like ear tagging (Allen et al., 2008). This study adds to the growing body of evidence supporting the viability of biometric identification in animal identification.

These studies, present the rapid development in cattle identification using biometric patterns rather than traditional methods like ear tags, human marks, etc.; it shows the potential of using various biometric features and advanced technologies to build up reliable and efficient solutions which could ensure better welfare in livestock management.

2.2.2 Other Combined Features

Other than using biometric features in animal detection, some researchers have developed a detection model based on cattle's individual behavior. Liu Liping et al. (2019) patented a four-axis system based on color block identification, which, although not specific to cows, divides objects into color categories and uses a machine-learning algorithm for real-time tracking. This method can potentially help build applications in cattle identification and health monitoring and provide a cost-effective and efficient solution.

Jian-Ping Wang and her team use Open Pose Mask R-CNN on cattle identification by utilizing open pose to extract skeletal features. They have divided the cattle body into five parts. Optimized the number of convolutional lays of Mask R-CNN's ResNet101 backbone and combined the Convolutional Block Attention Module (CBAM), for improved performance and achieved high accuracy rate of 90.2% and 92.1% on crossed subjects and views which indicate their model has strong generalization capability (Wang et al., 2023).

2.2.3 Distinctiveness of the Cattle Back Patterns

Besides those general characteristics mentioned above, cattle have a unique pattern only belongs to this species. In 1999, Grosz and MacNeil identified the genetic loci which responsible for the distinct coat pattern in cattle, and they have located the "spotted" Locus on bovine chromosome 6, this discovery highlights the genetic basis for the identical variability in cattle back patterns (Grosz & MacNeil, 1999). The unique marking and color patterns on the back of cattle are just like human fingerprint and DNA, which could play an important role in cattle identification and management.

2.3 Algorithm in Animal Detection

2.3.1 YOLO Algorithm

One of the most significant advances in the field of animal detection is the application of the YOLO (You Only Look Once) algorithm. YOLO is a deep learning algorithm known for its real-time object detection capabilities, which are widely used around different subjects, including data capture and online monitoring. Unlike the traditional method of scanning an image multiple times, YOLO views the entire image in one pass, making it very fast and efficient. According to the paper by Senbagam and Bharathi (2024), they've implemented the YOLO algorithm for detecting and identifying wild animals under varying weather conditions, which shows that YOLO is also good when capturing images in an unstable environment. Another study by Prashanth Kambli et al. (2022) used the YOLO algorithm to detect pest animals in agriculture. This study demonstrates the effectiveness of YOLO in real-time detection, which is essential for preventing incidents involving farmers. The algorithm, which uses deep learning to track movement and identify objects, is particularly useful in environments where pest animals are common, such as tea and coffee plantations. Furthermore, Kambli et al. (2022) also highlights the importance of using YOLO in automated systems to detect pest animals in agricultural Settings. The study highlights how YOLO can aid biological research and improve farmer safety by providing a reliable way to monitor potentially dangerous wildlife. Those advantages make YOLO particularly popular in applications that require real-time processing under complex conditions, such as surveillance and wildlife monitoring.

The YOLO algorithm is undoubtedly one of the most important foundations for computer vision training and biological monitoring studies. The YOLO algorithm used by the research team from the University of Bristol to frame out different cattle from multiple surveillance videos of the farm to help us move to the next step of machine learning analysis, which becomes the dataset of this dissertation.

2.3.2 ResNet50 Architecture, Advantages, and Applications

ResNet50 is a powerful convolutional neural network (CNN) that builds upon the residual learning framework (He et al., 2016). This network is composed of 50 layers, including a series of residual blocks that allow the network to learn identity mappings. Each residual block contains multiple convolutional layers with shortcut connections that bypass one or more layers. It can solve the vanishing gradient problem effectively and enable the network to be more capable of learning complex tasks without degrading performance as depth increases.

The key advantage of ResNet50 is its ability to improve the learning efficiency and generalization of deep networks. ResNet50 is well suited in contexts of animal classification tasks, which often involve complex patterns. It could capture complex details and variations in animal images like fur texture, colors, and different poses and has been effectively used in several studies to improve accuracy in recognizing animal species from images. Shows its ability to learn discriminate features with high-dimensional data:

Transfer Learning and Fine-Tuning: ResNet50 is pre-trained on large datasets like ImageNet and then fine-tuned for specific tasks. In studies like Image-based animal recognition based on Transfer learning (Collazos Huertas et al., 2021), ResNet50 significantly improved classification accuracy over traditional CNNs due to this approach.

Robustness: ResNet50 is adept in its robust architecture at handling variabilities in animal datasets, such as variations in pose, lighting, and occlusion. In the study Improving the classification accuracy of fishes and invertebrates using residual convolutional neural networks (Zhou et al., 2023), they used ResNet50 to classify fish and invertebrates and made comparisons with other models; the result shows ResNet50 consistently outrange other models, highlight its robustness in biological classification tasks.

2.3.3 Other Algorithms in Development of Cattle Recognition

In addition to the YOLO and ResNet50, many advanced learning models have also been applied, and much effort has been put into animal detection - for example, Efficient-Net, Siamese, Triple-Stream, and RCNN.

In the article *"Cattle Detection Occlusion Problem"* (A. Mendu et al., 2022), the team explores how drones and deep learning algorithms can be used to detect and identify herds over large areas. To address the problem of occlusion of cattle in images, they evaluated and compared four state-of-the-art target detection algorithms: YOLOv7, RetinaNet with ResNet50 and EfficientNet backbone networks, and Mask R-CNN. The experimental results show that in all test scenarios, YOLOv7 performs best, and its accuracy reaches 0.612. The performance of these methods on both RGB and thermal imaging datasets was compared to improve the accuracy of cattle detection in occlusions.

In addition, EfficientNet has also been effectively used to solve cattle recognition problems and improve livestock management. Several approaches have been applied to this topic. For example, a study by Zhang et al. (2023) combined EfficientNet with DeepOtsu for individual cow identification and achieved a 98.5% identification accuracy (Zhang et al., 2023). At the same time, Yin et al.(2020) integrated EfficientNet with LSTM for recognizing dairy cow behaviors like lying and feeding and achieved 97.87% in behavior recognition and outstanding the existing model by 4.25% (Yin et al., 2020).

Triple-stream network is a relatively new method in animal detection. It compares the inputs for the distance to an anchor, giving a positive and a negative example. In 2022, Yang Yang et al. introduced the triple-stream network in real-time cattle interaction recognition. It mentioned that this method could help combine visual, geometric, and semantic features (prior knowledge of individual actions and interactions) to recognize cattle interactions effectively, even with limited labeled data (Yang et al., 2022).

2.3.4 Summarize of Animal Detection Algorithms

Below is a chart which summarize the characterize for above algorithm and their ability:

Algorithm	Summarize
YOLO	Excellent performance in speed and accuracy for real-time detection in the range of 5 to 160FPS. Using an extended ELAN architecture for new label assignment.
ResNet50	Is a convolutional neural network excelling in image classification and feature extraction tasks. It uses a deep 50-layer architecture with residual blocks to mitigate the vanishing gradient problem, making it suitable for high-accuracy image recognition and computer vision tasks.
EfficientNet	A convolutional neural network known for its efficiency in accuracy and computational resources. It scales depth, width, and resolution uniformly using a compound scaling method, resulting in superior performance with fewer parameters. Suitable for a wide range of image classification tasks requiring speed and accuracy.
Mask R-CNN	It is a convolutional neural network that is good in image segmentation and instance segmentation. Use ResNet-50 as a base-stone network. Suitable for complex object detection.
Triple-Stream	It is good to compare the inputs for the distance to the anchor with given a positive and a negative example. Often used in the context of triplet loss.

Table 1 Summarize of Animal Detection Algorithms

2.4 Gaps and Our Directions

From the comprehensive analysis of the existing literature, the current research on cattle recognition mostly focuses on high-precision recognition technologies, such as iris recognition, muzzle print recognition and limb combination model. These methods usually require high resolution of the acquisition device, consume a lot of computational resources in the process of deep learning, and the learned models tend to focus on too fine feature details. It is worth noting that although studies have shown that the patterns on the back of cattle is as distinct as human fingerprints, the research on bovine identification using the color block on the back of cattle has not been widely carried out.

Therefore, the primary focus of this study is to develop an approach using color blocks on cattle’s backs as distinctive features for identification, distinguishing each individual from the group. The advantage of utilizing this feature lies in the uniqueness of cattle’s back pattern, because it’s not only as unique as human fingerprint but also the size is big enough which does not require a high-resolution equipment. This approach could practically benefit in practical use that many farms has only limited monitor resources and may not support high-resolution outputs or substantial computational power. Using this approach could also benefit and improve animal welfare other than using traditional methods like marking or injecting.

The secondary goal of this research is to evaluate, compare, and rationalize various models to find effective methods for minimizing computational resource consumption while ensuring the accuracy of recognition based on cattle back color blocks. This involves optimizing algorithmic efficiency, simplifying model complexity, and developing deep learning techniques suitable for single channel or low-resolution inputs. These strategies aim to make this recognition technique viable even in environments with constrained resources.

3 Methodology

The primary objective of this dissertation is to accurately identify cattle based on their distinct back color block patterns within a general environment by optimize the effectiveness of this pattern and exploring the model parameter. As an additional objective, this study seeks to minimize computational resource consumption while maintaining cattle identification precision.

3.1 Data Collection and Preparation

3.1.1 The Cows2021 Dataset

The dataset we use for building and testing algorithms is a public dataset provided by the University of Bristol in the paper *"Towards Self-Supervision for Video Identification of Individual Holstein-Friesian Cattle: The Cows2021 Dataset"* (Gao et al., 2021). This dataset is a collection of 186 Holstein-Friesian cattle, including 10,402 RGB images (1280 × 720 pixels) and 301 videos (each 5.5s long at 30fps).

The dataset focuses on capturing the breed's color block patterns for individual identification and consists of three parts: (a) Detection and localization, (b) Identification, and (c) Weights.

Figure 1 Example of Cattle Images (Source: Gao et al., 2021)

The data we are using is extracted from (b) identification. This part has undergone preliminary processes, including:

Target detection: Generate an oriented bounding box using the YOLO algorithm to accurately detect and isolate the body part of each cow in the images.

Identification: Manually label and identify each cow, assigning a unique identification number to 13,784 detected cattle instances to facilitate individual recognition. Note that the actual number of cattle used for identification was 182, as four all-black cattle were excluded from the identification study.

Self-supervised learning: The team has leveraged the sequential information in the videos, using a tracking-by-detection to general tracklets and employed a contrastive learning method to strengthen the identity recognition model further. At the same time, the images are proceeded from 1280x720 pixels to 500x220 pixels due to reduction of environment part.

Thanks to the team from the University of Bristol, who completed above initial steps; we are now proceeding to the next stage.

3.1.2 Data Cleaning

After thorough review of the dataset, we found it organized into three categories: (a) Detection and Localization, (b) Identification, and (c) Weights. We focused on 'Identification' segment which is divided into 'Train,' 'Test,' and 'Videos' directories. The directories of 'Train' and 'Test' contain various numbers of cow images, systematically sorted into distinct folders.

The 'Train' directory contains 336 folders, which is exceeding the count of 186 target cattle initially outlined by the researchers. In addition, there are multiple sub-folders within the same folder, such as folders labeled (0,1,2) under folder No.5, with each representing different cattle. According to section 2 of the 'Dataset Cows2021' publication, their goal is to develop a self-supervised learning method to automatically identify and distinguish individual cattle by utilizing temporal information from video data. Therefore, the training dataset is designed and structured to reflect a natural flow of actual farm work, since sometimes multiple cattle can appear at the same time.

In the 'Test' dataset, 182 cattle were identified for use, with four cattle (numbered in ‘Test’ should be 54, 69, 73 and 173) were not included in the annotated data because they lacked distinct white markings for valid identification. Additionally, the folder numbered 179 is empty. As a result, only 181 cattle could be visualized. This resulted in approximately 8,700 pictures, each measuring around 500x220 pixels.

After a brief check of the 'Test' dataset, we did not find duplicate cattle with different IDs, and there are no two cows sharing the same ID. This is important because it helps our model learn correctly without getting confused. Due to the inconsistency of labels, the recognition accuracy of the model will be significantly reduced. The model may confuse that the same individuals under different numbers when learning features and making predictions, which would resulting in a high error rate.

3.1.3 Data Handling

To optimize processing, we converted the image format from 500x220 pixels to a 64x64x3 pixel format and stored them in CSV files.

This decision was driven by the limitations associated with CSV file formats and the need to retain color information which is crucial when trying to identify patterns or textures related features. This adjustment not only ensures compatibility with our data processing tools but also enhances the efficiency of our training operations while keeping rich details of the data.

The CSV file is made up of these data: the first 12288 columns of data are 64x64x3 pixel values, and the last column is the CowID labeled by the name of their folder and the first row is the label of Pixels and the CowID.

3.1.4 Data Segmentation

To ensure a comprehensive evaluation of our model, we break down our dataset into different segmentation to aligns them with best practices using a multi-step process:

1. Initial Split Using Stratified Shuffle Split:

We used **StratifiedShuffleSplit** to divide the data into approximately 70% for training (including future transfer learning data) and 30% for temporary data (validation and test). This method ensures that the distribution of classes (individual cattle) is preserved in each split, which is crucial for the model accuracy and generalization.

2. Split of Temporary Data:

The Temporary data are equally divided into Validation and Test sets (each about 15% of the total data). This split maintains the representative sampling ensured by the initial stratified split.

3. Extraction for transfer learning:

A 10% of the original training data was further divided for transfer learning subset. This subset allows us to explore the model's ability to adapt to slightly different tasks without significantly impacting its primary learning ability.

Final Data Distribution:

- ✧ Training set: 62.8% (5461 samples)
- ✧ Validation set: 15.0% (1300 samples)
- ✧ Test set: 15.0% (1301 samples)
- ✧ Transfer learning set: 7.0% (607 samples)

Each dataset has 181 label dimensions, corresponding to the number of unique cattle in our study. This Segmentation method ensured a balanced representation across all splits, facilitates effective hyper-parameter tuning, enable credible model validation and provides a separate set for exploring the transfer learning abilities.

3.1.5 Data Augmentation and Image Preprocessing

To enhance the model’s generalization and recognition capability, we implemented a series of data augmentation and preprocessing techniques. These are divided into two main categories: data augmentation and image preprocessing.

1. Data Augmentation

The image is augmented before training, with utilized Keras' **ImageDataGenerator** to implement the following strategies:

a) Rotation: Set **rotation_range=90**, this allowing the images to rotate randomly between -90° and +90°. This simulates the natural environment which cattle might appear in various angles, helping the model learns features from different orientations.

b) Translation: By setting width_shift_range=0.1 and height_shift_range=0.1, images can move randomly for up to 10% of their width and height, in a horizontal or vertical directions. This makes sure the model could identifying cattle at different positions within the frame.

c) Zoom: Set a zoom_range=0.1, allows the image to be randomly zoomed in or out by up to 10%. Which enables the model to recognize cattle of different sizes or at different distances.

d) Shear: The shear_range=0.1 applying a light shear transformation to the images. This is simulating image distortions that might occur due to changes in viewpoints.

e) Flip: By using horizontal_flip=True, image is randomly flipped horizontally so that the model can recognize images with left and right symmetry.

f) Fill Mode: A fill_mode='nearest', using a nearest-neighbor interpolation to fill any blank areas that might be created during above transformations.

After making these changes, we ensure there are multiple images of the same animal taken from different angles and positions. This variety helps our model learn more effectively by seeing the cattle in many different situations.

2. Image Preprocessing

In addition to data augmentation, we implemented In addition to data augmentation, we implemented several different image preprocessing methods to explore to compare the effects of different preprocessing methods:

a) **Original Images**: Maintain the original images unchanged, only applying normalization (pixel values divided by 255).

b) Grayscale Conversion: Using **OpenCV's cv2.cvtColor** function to convert RGB images to grayscale, reducing data volume while retaining primary features. To Explore the impact of color information on recognition accuracy.

c) Resolution Reduction: Using **OpenCV's cv2.resize** function to reduce image resolution from the original 64x64 to 32x32 in order to test the effect of lower resolution on model performance and computational efficiency.

Output results using different instruction:

Original Image: 64x64x3 RGB images.

Grayscale Image: 64x64x1 grayscale.

Half-Resolution: 32x32x3 RGB images.

3.2 Experiment Setup

3.2.1 Model Selection and Customization

Based on our literature review, we evaluated the effectiveness of different models in feature recognition project especially for cattle identification. Our objective requires a learning network proficient in image classification and feature extraction. Through our analysis, we determined that ResNet50 is particularly well-suited for this task due to its robust performance in image recognition tasks and the capability to extract features with high accuracy.

We selected ResNet50 as our primary model. To make sure it meets our needs, we made several parameter adjustments. These modifications ensure that the model not only excellent in general image classification, but also optimized for recognizing cattle based on their distinctive back color patterns.

For comparison, we also implemented the EfficientNetB0 and a custom sequential to evaluate the trade-offs between model complexity, accuracy and computational resource requirements.

3.2.2 Model Customization

We made several important modifications to the base architecture:

a) Removal of the Top Layer: For ResNet50 and EfficientNetB0 we removed the top fully connected layers (include_top=False) to replace them with layers better suited to our cattle identification problem than the original ImageNet 1000 class problem.

b) Input Shape Adaptation: The input shape is adjusted to (64, 64, 3) to fit the image size used in this project. Adjusting the input shape allows the model to work efficiently with smaller images, which is important for specific deployment environments where high-resolution images might not be available.

c) Grayscale Input Handling: For grayscale images (64, 64, 1), we added a convolutional layer to convert the single-channel input to a three-channel input. This modification allowing us to use pre-processed images.

d) Global Average Pooling: After the base model, we applied Global Average Pooling to reduce the spatial dimensions.

e) Dense Layers: We added a dense layer with 128 units and ReLU activation for additional feature learning.

f) Dropout Layer: A dropout layer with a dropout rate of 0.5 is included to reduce overfitting. By randomly setting the output features of the dense layer to zero during training, dropout forces the model not to rely on any small set of neurons, thus enhancing its generalization capabilities.

g) Output Layer: The final layer is a dense layer with softmax activation, where the number of units equal to the number of distinct cattle in our dataset.

3.2.3 Hyperparameters and Training Process

We conducted comprehensive experiment to optimize the training process:

a) Batch Size: We tested batch sizes of 32 and 64, finding that a batch size of 32 provided a more stable learning behavior. With a batch size of 64, the model shows an unstable learning behavior which likely due to the reduced number of samples in each batch, affecting the gradient estimation.

b) Learning Rate: We implemented a **ReduceLROnPlateau** callback to dynamically adjust the learning rate, with an initial rate of 0.001, a reduction factor of 0.2, and a patience of 3 epochs.

c) Early Stopping: To monitor and ensure the model's learning quality, we implemented a model check-pointing to record the weights of the best-performing model on validation accuracy. Based on an initial experiment running for 1000 epochs, the curve drives sharply and converges closely for around the 50th epoch; it shows there are no significant improvements after this point. Based on this observation, we decided to implement an early stopping mechanism set at 60 epochs since the curve stabilized after the 50th epoch.

d) Model Weights Retrieve: We saved the best model weights based on validation loss, ensuring we retain the most effective model configuration to compare the model training speed base on the epoch of best model weight.

e) Logging: We implemented a CSV logging method to record our training and validation metrics across the experiment, for reload the model and analysis the learning process.

3.3 Performance Evaluation Methods

The models are trained using Adam optimizer with categorical crossentropy loss. The training and experimenting process uses above data argumentation method to enhance the model generalization ability.

In this dissertation, we use below metrics evaluate the performance of above models:

1. Validation Loss & Validation Accuracy

These metrics are important during the training process as they monitor the model's performance on unseen data. We use these data, comparing with Training Loss & Accuracy to understand the progress of training.

2. Test Loss & Test Accuracy

The test lost and test accuracy helps evaluate the model's performance on a completely separate dataset that haven't been used during the training or validation phases. Help us to see how well the model performance.

3. Transfer Learning Accuracy

The transfer learning accuracy is using a different image set to test and predict the cattle number using another image rotation method, which retains the inherent patterns of each cow but rotates from a random direction for data augmentation. They help in assessing the model's ability recognize unseen images, which is closely mimic the real-word scenarios.

4. Best Epoch

This data is recorded to help us understand when the model reaches its best performance under the trained epochs. Can help us understand which model completes the training first and prevent from unnecessary training.

5. Training Time/Epoch:

This focuses on comparing the training speed per epochs. This data is important for selecting a model that is not only high-performance bust also resource efficient, which could benefit practical use.

The training curves and metrics are recorded in file.

4 Results and Analysis

This chapter presents the experimental findings of the dissertation, providing a thorough and detailed evaluation of the results. The analysis includes both quantitative and qualitative aspects, reflecting the strictness required in the field of cattle identification using deep learning models.

Before delving into the results, a brief recap of the experimental setup is provided below:

Models Used: ResNet-50, EfficientNetB0, and a custom Convolutional Neural Network

Dataset: Cows2021 Dataset(), processed and augmented as described in the methodology.

Performance Metrics: Validation loss, validation accuracy, test loss, test accuracy, transfer learning accuracy and training time per epoch

Tools and packages: TensorFlow, Keras, Pandas, NumPy, OpenCV, sklearn, Matplotlib, os

Hardware: NVIDIA GeForce RTX 4070TI, Intel Core i7-13700KF

Image setting: image resolution and color setting

4.1 Deep Learning Models Performance Comparison

The training data were recorded as shown below. We are going to analysis the model performance based on these figures:

- **Validation and Test Accuracy:** Present the accuracy of each model on the validation and test datasets. Use tables and graphs to compare performance across models.
- **Validation and Test Loss:** Discuss the loss metrics, highlighting differences in convergence rates and overfitting tendencies.

- **Transfer Learning Accuracy:** Prediction accuracy based on 300 random rotated images from 607 total images with transfer learning dataset
- **Training Time:** Compare the training time per epoch for each model, discussing computational efficiency.

4.1.1 Comparison Between Different Model Used

We conduct an experiment running for different models in 1000 epochs without early stopping for testing a baseline of their performance and pick up which model we should dive further. Batch size was set as 32.

The model performance metrics of each model is recorded in the table below. A brief analysis reveals their different advantage and disadvantages:

Model	Val Loss	Val Accuracy	Test Lost	Test Accuracy	Transfer Accuracy	Best Epoch	Seconds/Epoch
Custom	0.30009	0.91846	0.32737	0.92929	0.956	986	5
ResNet50	0.16433	0.97462	0.12824	0.97925	0.982	976	7
Efficiency	0.32254	0.93385	0.35826	0.94081	0.936	343	14

Table 2 Model Performance Summary

Observations:

Custom Model: The custom model has the fasted training speed at 5 second per epochs. But the metric it exhibits the lowest Val and Test accuracy among the three models. It also reached its best performance at the latest epoch (986), indicating a slower convergence.

EfficiencyB0: Although the EfficiencyB0 has the lowest training time per epoch (14 seconds), it achieves its optimal performance much earlier at epoch 343. While it has a better Validation and Test accuracy compared to the Custom Model, it's Transfer Learning accuracy is slightly lower, suggesting it may be less effective at recognize feature for the training data.

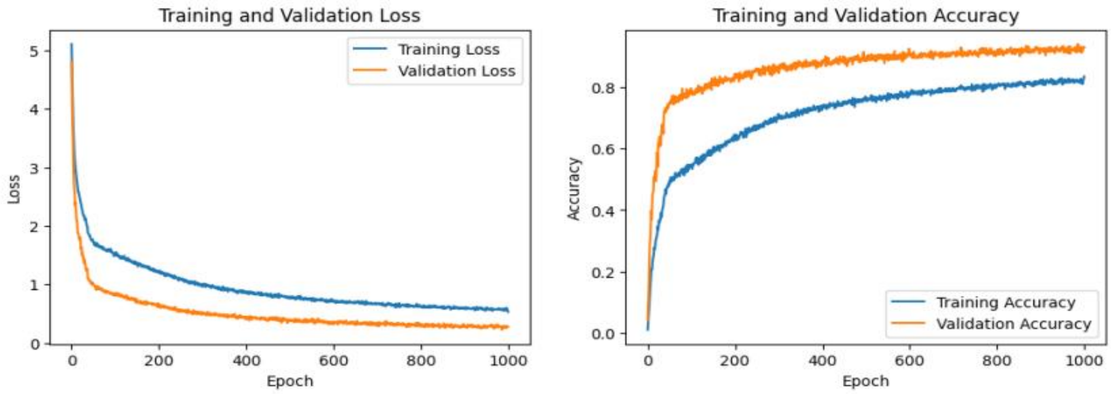
ResNet50: The best performer over three models. ResNet50 offers the highest Transfer Learning accuracy of 0.982, and the best Validation and Test Accuracy rate. It also shows a lowest Validation and Test Losses. Those clearly establish its superior performance across our project target. Although it takes longer for training time(8 seconds) per epoch compared to the Custom model(5 seconds), the overall efficiency and accuracy are relatively better.

Each Model shows their unique strengths and trade-offs between training speed, accuracy, and the ability to learn from data, which should be considered when selecting models for specific purposes.

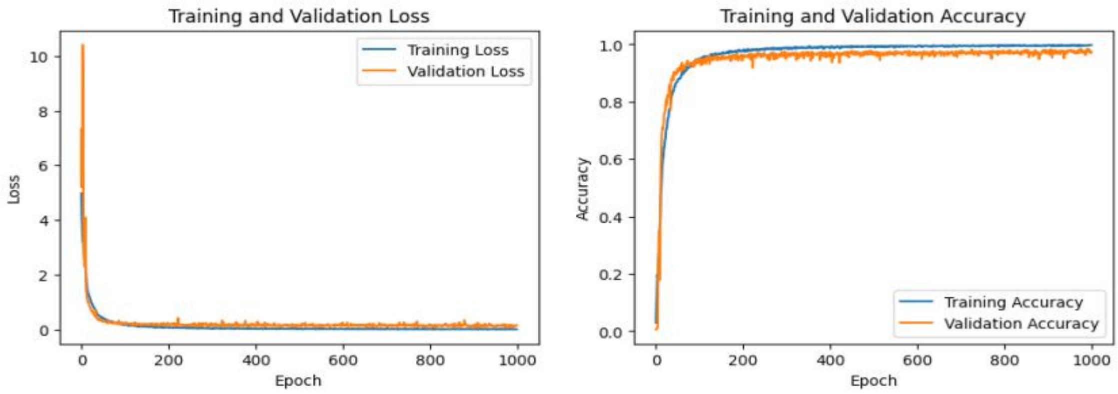
4.1.2 Comparison between Training Curves

Besides assessing general performance, we also analyzed the training curves for all three models, which show significant differences in behavior and outcomes that influence our choice.

Custom Model



ResNet50



EfficiencyB0

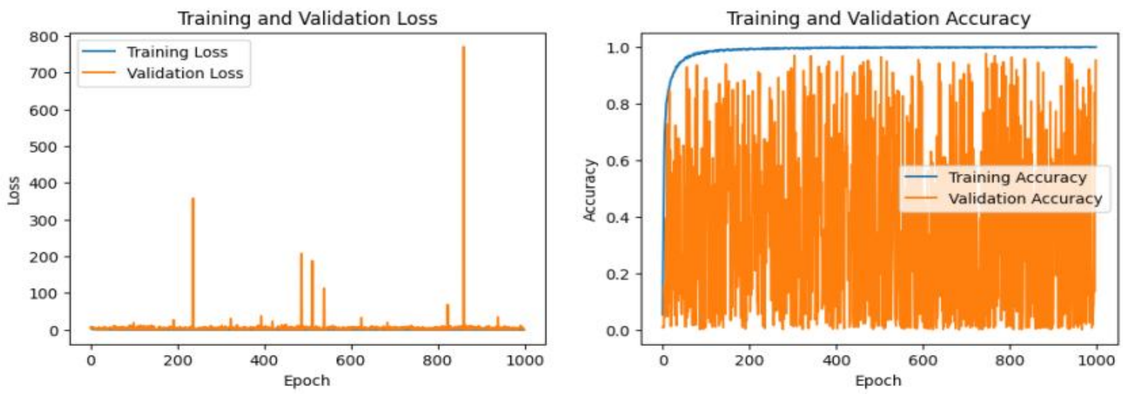


Figure 2 Different Models' Training Curves

Each model displays distinct training dynamics:

The **Custom Model** has a relatively stable curves, the loss decreases smoothly and the accuracy increases consistently, indicating that the model is learning gradually without over-fitting. Shows its robustness and making it suitable for scenarios where reliability is important.

ResNet50 is excellent in performance, with both training and validation losses converge rapidly and the validation accuracy closely tracks the training accuracy. This fact suggesting that the model is not only learning well but also generalizing well to new data.

The **EfficiencyNet** Model exhibit some instability, as seen in the wild fluctuating validation loss and accuracy. These fluctuations could indicate issues like gradient explosion or inadequate parameter tuning. The sharp twists in validation loss suggest that the model might be too sensitive to variations in the training data or settings although we have introduced Learning Rate Reduction. We have tested the EfficiencyNet in two different settings, with all of them shows a instability trend. The EfficiencyNet, while theoretically efficient, may require carefully tuning to against the instability.

4.1.3 Conclusions on Model Performance

Based on above analysis and observation, **ResNet50** stands out to be the best choice for our project and here is a summary of why:

Superior performance:

- Highest Transfer Learning accuracy (0.982)

- Best Validation and Test Accuracy rates
- Lowest Validation and Test Losses

Efficient Learning:

- Rapid convergence of training and validation losses
- Validation accuracy closely tracks training accuracy

Generalization:

- Good performance on both training and validation data indicates strong generalization capabilities

Stability:

- Consistent performance without the instability seen in EfficiencyNet
- While ResNet50 has a slightly longer training time per epoch compare to the custom models, it's overall efficiency and accuracy make it the superior choice. The EfficiencyNetB0, needs requires more carefully tuning to overcome the instability needs further development. Which made ResNet50 become the optimal model for our project.

4.2 Impact of Image Preprocessing Methods on Model Performance

After identifying the best model for our project, we conduct a further experiment to analysis of how the imaging prepossessing techniques effected both the learning efficiency and model accuracy. Building upon our baseline model, which utilizes images of 64x64x3 pixels, we implemented and evaluated two additional prepossessing strategies. The training parameters, including learning rate and optimization algorithm, remained constant to ensure comparability.

4.2.1 Grayscale Conversion in Result

We explored the efficacy of grayscale images compared to the original RGB format to assess the importance of color information in our cattle identification task. Our base RGB images were converted to grayscale by reducing the input channels from 3 to 1, and we adjusted the model architecture to accept single-channel inputs for grayscale images. The result of shows in Table2:

Image	Val Loss	Val Accuracy	Test Lost	Test Accuracy	Transfer Accuracy	Best Epoch	Seconds/Epoch
Baseline (64x64x3):	0.17438	0.95923	0.20753	0.95772	0.953	171	7
Grayscale: (64x64x1)	0.31818	0.92462	0.24462	0.93928	0.931	176	7

Table 3 RGB vs Grayscale Image

Observations:

Based on our results, color does play a significant role in cattle identification. The transition to grayscale imagery resulted in a slight decrease in validation accuracy (3.46%) and test accuracy (1.84%). However, the transfer learning accuracy only dropped by 2.2%. Thus, while color information contributes to cattle identification accuracy, grayscale models still retain most of the essential features. The higher validation loss in the grayscale model (0.31818 vs 0.17438) indicates that the model is less confident in its predictions, which is possibly due to the loss of color information.

We were wondering how much color affects the recognition of cattle color blocks since they are said to have unique patterns. Is it possible to convert them to black and white and still recognize them? Our results confirm that these patterns remain largely distinguishable even in grayscale. The small drop between full-color and grayscale accuracy rates (especially in test and transfer learning accuracy) indicates that the model still has a good ability to identify individual cattle with high accuracy using only grayscale images.

4.2.2 Resolution Reduction in Result

We also investigated the impact of image resolution on model performance by conducting experiments with reduced resolution inputs. We compared our baseline 64x64x3 pixel images with a half-resolution variant (32x32x3 pixels), and two experiments were conducted with the half-resolution images: one with a batch size of 32 and another with a batch size of 64.

Image	Val Loss	Val Accuracy	Test Lost	Test Accuracy	Transfer Accuracy	Best Epoch	Seconds/Epoch
Baseline (64x64x3):	0.17438	0.95923	0.20753	0.95772	0.953	171	7
Half (32x32x3)	0.36564	0.92692	0.33445	0.93620	0.857	347	7
Batch 64 (32x32x3)	0.30000	0.92538	0.32519	0.92775	0.903	429	3~4

Table 4 Full vs Half Resolution Image

Observations:

The half-resolution experiments revealed a trade-off between model performance and computational efficiency:

With a batch size of 32, the half-resolution model shows a 3.23% decrease in validation accuracy and a 2.15% decrease in test accuracy compared to the baseline. Notably, it required approximately twice the number of epochs (347) to reach its optimal weights, and the transfer learning accuracy dropped significantly by 9.6%. These observations prompted us to conduct an additional experiment with a batch size of 64, aiming to achieve smoother learning dynamics.

Increasing the batch size to 64 for the half-resolution model yielded mixed results. While it improved the transfer learning accuracy (0.903 vs 0.857), it slightly decreased the test accuracy (92.775% vs 93.620%). As a trade-off, the learning process was slower, with the model reaching its best weights at a later epoch (429), effectively reducing the learning speed by about 50%.

Both half-resolution models exhibited higher validation losses compared to the baseline, suggesting less confident predictions.

4.2.3 A Comparison Between Training Curves

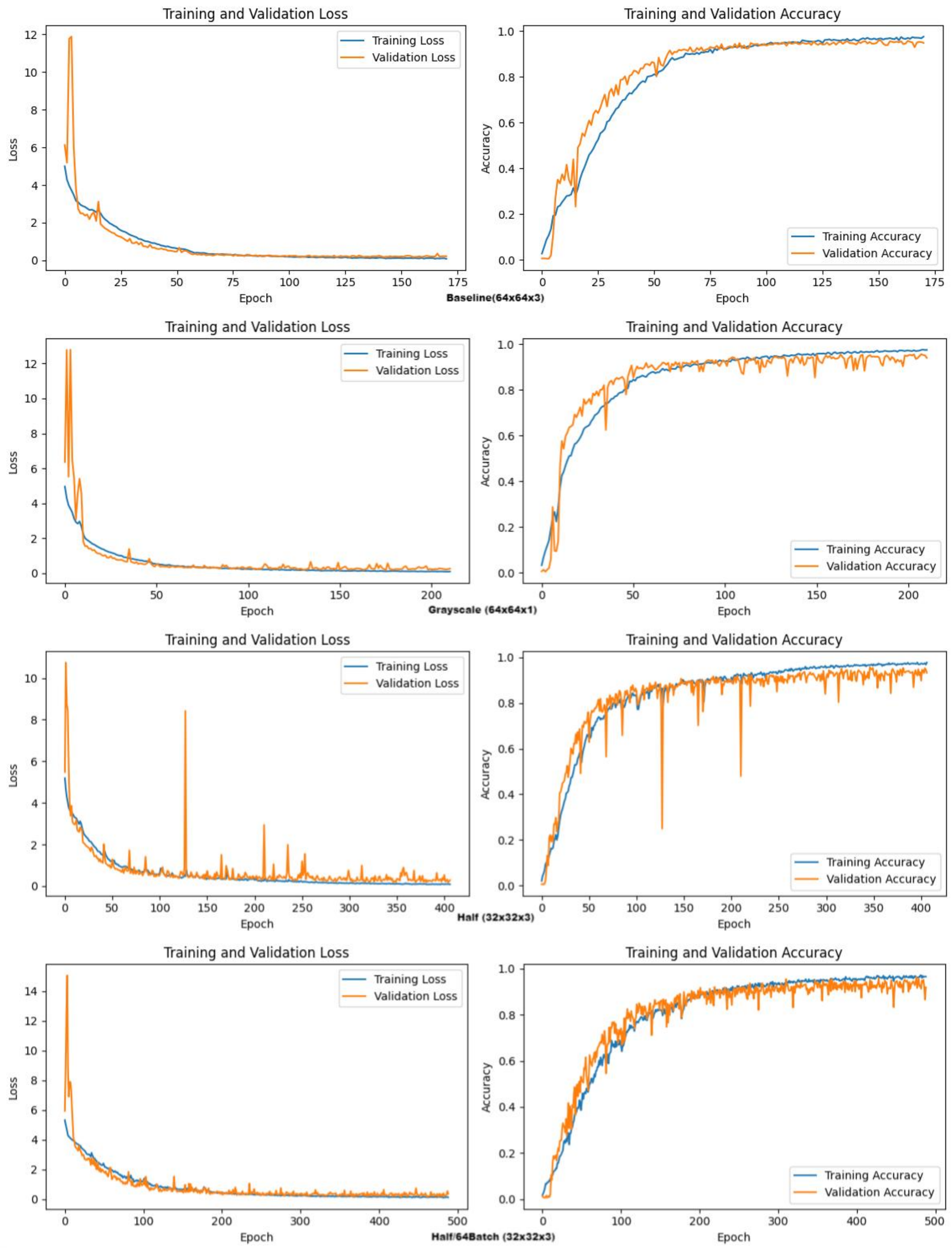


Figure 3 Training Curves for Different Image Processing Methods

The learning curves provide further insights into the models' behavior:

- The **Baseline model** (Image 1): demonstrates smooth and consistent learning progress for both training and validation sets.
- The **Grayscale model** (Image 2): exhibits more fluctuations in validation accuracy compared to the baseline, especially in later epochs, suggesting less stability in learning.
- Compared to the grayscale images, the training curves for **Half-resolution** models (Images 3 and 4) show increased instability in validation accuracy, especially in the one with **32 batch size** (Image 3). There are some sharp spikes before epoch 250, which shows a very poor quality in some of the validation. This might cause by some challenging sample in the validation or the poor quality for learning at that epoch. By increasing the **batch size to 64** (Image 4), we let the model learn more smoothly and the sharp spikes disappeared. However, small spikes still exist, indicating that lower resolution does affect the training quality.

These observations underscore the complex interplay between image resolution, batch size, and model performance, highlighting the need for careful optimization in resource-constrained scenarios.

4.2.4 Conclusion

In In this experiment, the influence of image preprocessing technology on the performance of the cattle recognition model is fully discussed. From a baseline of 64x64x3 RGB images, we explored two main preprocessing strategies: the grayscale conversion and the resolution reduction.

According to the Grayscale conversion experiment, compared with our baseline model, the validation, test, and transfer learning accuracy of the grayscale model all decreased slightly by 1.8% to 3.5%. However, transfer learning accuracy still has a good result of 93.1%. It shows that although color information plays a vital role in cattle recognition, the grayscale images (64x64x1) could still retain most of the key features. This indicates that the grayscale model may help improve the efficiency of data storage, transmission, and processing while maintaining good recognition accuracy.

Resolution reduction experiments reveal the trade-off between model performance and computational efficiency. Reducing the image resolution to 32x32x3 pixels resulted in a slight decrease in validation accuracy and test accuracy. However, the impact of lowering the transfer learning accuracy was more significant by approximately 10%. An increased batch size of 64 can improve the model's learning stability and transfer learning performance, but it will cost more training epochs. For this reason, lowering the resolution of cattle images may not be a good choice when considering speeding up the training due to its instability and inefficiency.

By analyzing the learning curves, we observe the influence of preprocessing methods on the learning dynamics of the model. The baseline model shows the most stable learning process, while the grayscale and low-resolution models show varying degrees of fluctuation, especially in terms of validation accuracy.

These findings highlight the importance of optimizing image preprocessing strategies in resource-constrained scenarios. According to the specific application requirements, the trade-off between accuracy and computational efficiency can be made to choose the appropriate pretreatment method. Future research could explore more refined preprocessing techniques, such as adaptive resolution scaling or selective color channel retention, to further optimize cattle identification systems' performance in various deployment environments.

4.3 Error Analysis and Quality Control

4.3.1 Result in Misclassifications

Our Cattle recognition model shows promising results, but there are some areas for improvement. To further discuss the reason of common misclassifications for identify cattle, we move into an in-depth evaluation.

Confusion Matrix

Below is confusion matrix based on the prediction of the whole Transfer Learning Dataset (607 images). presents the confusion matrix based on predictions for the entire Transfer Learning Dataset, comprising 607 images. The matrix provides valuable insights into the model's performance across different categories.

Key observations:

Diagonal dominance: The concentration of dark blue points along the diagonal indicates the model's high accuracy across most categories. The darker shades represent a higher number of correct predictions.

Misclassifications: Each scattered red points off the diagonal represent a instance of misclassification. While relatively few, these points highlight specific categories where the model faces challenges.

Performance variation: The inconsistent intensity of blue along the diagonal further emphasizes how the model's performance varies across different categories, likely influenced by the uneven distribution of samples.

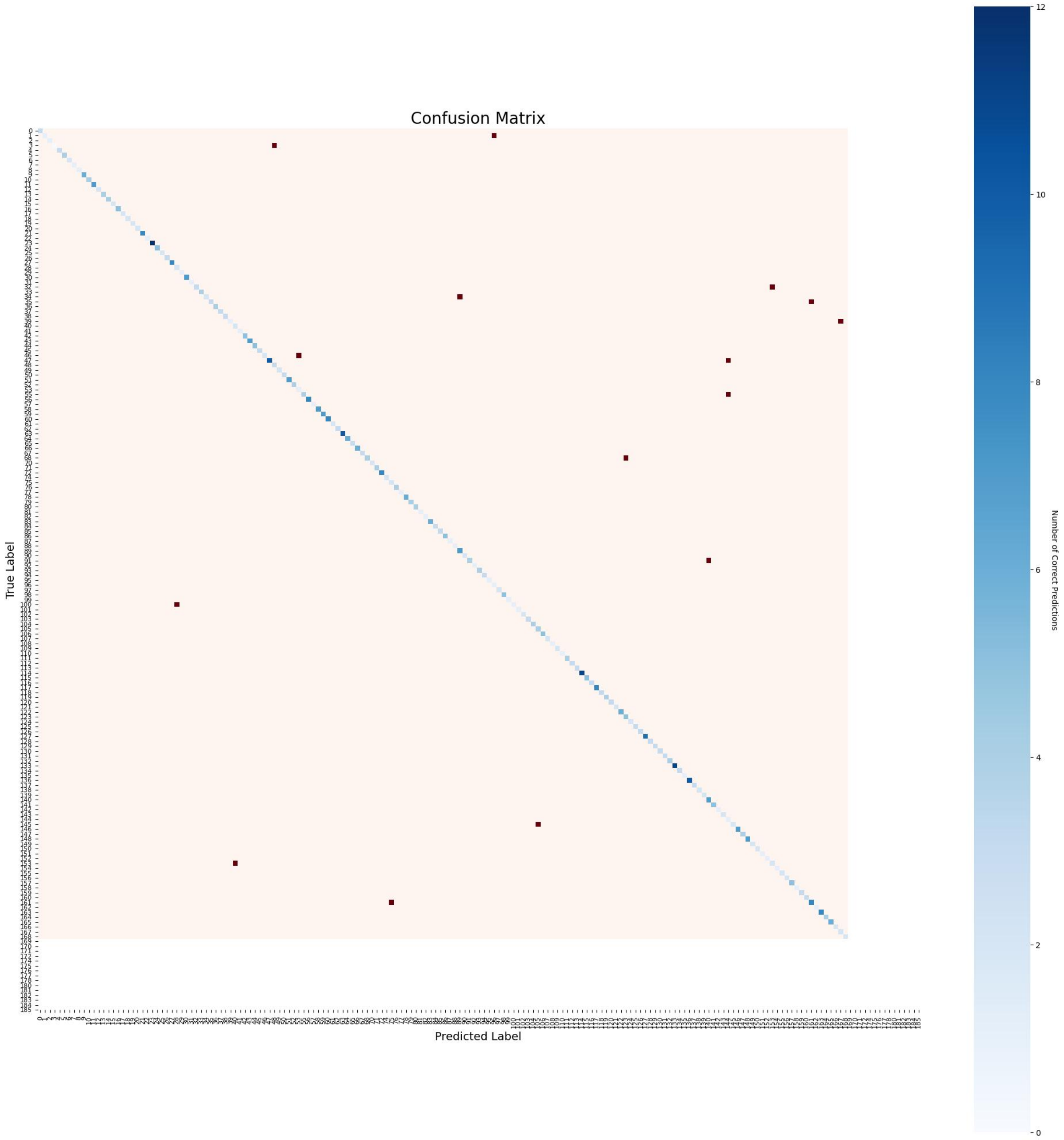


Figure 4 Confusion Matrix of Prediction Results

Misclassification Patterns Analysis

For an in-depth evaluation of error analysis, we print out all result for all Misclassification example and Quality Control Indicators to find out key error patterns and quality issues.

All Miss-classifications		
Label	Predicted	Count
1	101	1
3	50	1
34	161	1
36	93	1
37	172	1
41	180	1
48	56	1
49	150	1
57	150	1
71	128	1
95	145	1
105	29	1
151	110	1
161	42	1
172	77	1

Bottom 15 worst performing classes:				
Label	Precision	Recall	F1-score	Support
175	0.667	1.000	0.800	2.0
74	0.667	1.000	0.800	2.0
42	0.667	1.000	0.800	2.0
148	1.000	0.667	0.000	3.0
48	1.000	0.667	0.800	3.0
36	1.000	0.667	0.800	3.0
29	0.667	1.000	0.800	2.0
41	1.000	0.500	0.667	2.0
1	1.000	0.500	0.667	2.0
158	0.667	0.667	0.667	3.0
55	0.500	1.000	0.667	1.0
98	0.500	1.000	0.667	1.0
102	1.000	0.500	0.667	2.0
147	0.333	1.000	0.500	1.0
3	0.000	0.000	0.000	1.0

Misclassification Frequency and Distribution

In the observed misclassification, there is no high frequency of specific errors. All the misclassifications occur only once each, suggesting that there is no systematic pattern of errors. Notably, both cattle with labels 49 and 57 were misclassified as cattle 150, prompt a further discussion into their pattern similarities.

In addition, misclassifications are distributed evenly across different categories, this indicate that the model does not exhibit bias towards specific categories and indicate of a generally robust classification capability of our model.

Quality Control Indicators

With only 15 unique misclassification instances, it indicates that the model performs well across most of the dataset. Thus, we are not doing specific discussion on these well- performed categories.

To match with the misclassification, we drag out the worst 15 performance category: Category 3 has a score of 0 for all indicators, which is a serious quality issue. Other categories such as 55, 98, 102, 147 also have varying degrees of quality problems. However, these five categories have low support of 1 or 2, suggesting that their poor performance metrics might be influenced by the small sample size. This limitation in the dataset could skew the perceived model quality and requires further consideration.

4.3.2 Analysis of Misclassification Patterns

Figure 5 Example of Misclassifications

Similar Overall Patterns

One of the main findings observed was that the models were not distinguishable between cattle with a similar overall black and white distribution. For example:

- Cattle 49, 57, and 150: All three cattle have white areas concentrated on the head and tail in similar position, with a back mainly covered by large black areas.

- Cattle 95 & 150, 151 & 110, 1 & 101, 41 & 180, 172 & 77: These pairs of cattle all have large (over 80%) black areas on their back, and some small white patches are similarly distributed.

This finding suggests that the model may rely too much on the overall black-and-white distribution ratio, but not focusing enough on the specific shape of these color blocks.

Similar local features

Some misclassification cases show that the model may be overly focused on certain specific features, but forget to look at other patterns:

- Cattle 71 & 128: They all have two black teardrop-shaped color blocks in the center of the body, with a similar distance near the spine. However, there is a triangle-shaped black pattern on 71, while 128 doesn't have.
- Cattle 161&42: Large complex black pattern in the center of the back which hard to describe. But closer to the neck area we can clearly see the difference.

This suggests that models may in some cases rely too heavily on similarity of a single feature, while ignoring differences in overall patterns.

Similar Edge Pattern

- Cattle 36 & 93: Although the overall pattern is irregular, they show some similarity in the edge of the black and white border. At first glance we even thought their pattern was the other way around.

Modular Pattern Recognition and Composition

In the case that our model misclassified cattle 49 and 57 as cattle 150, the differences and similarity of them are very interesting:

Tail pattern similarity: The upper part of the tail pattern for cattle 49 and 57 (especially when viewed from certain angles) closely resembles that of cattle 150. This similarity is particularly noticeable when looking at the area above the spine line.

Head pattern similarity: Cattle 57 and 150 have nearly identical head patterns, both showing an inverted Y-shape with a protrusion on the left branch. Cattle 49's head pattern differs slightly, showing an upright Y-shape with a thicker right branch.

This kind of misclassification might cause by the CNNs often learning to recognize patterns in a hierarchical and modular way, that they learned the edges or basic shapes in lower layers, then combine them in higher layers to recognize complicated patterns. This means that the model might already be learned to recognize various similar shapes of blocks across multiple cattle rather than seeing their body as a unique pattern.

In addition, CNN does have some degree of positional invariance that can then recognize patterns regardless of their location. Which could contribute to confusing on similar patterns in different orientations.

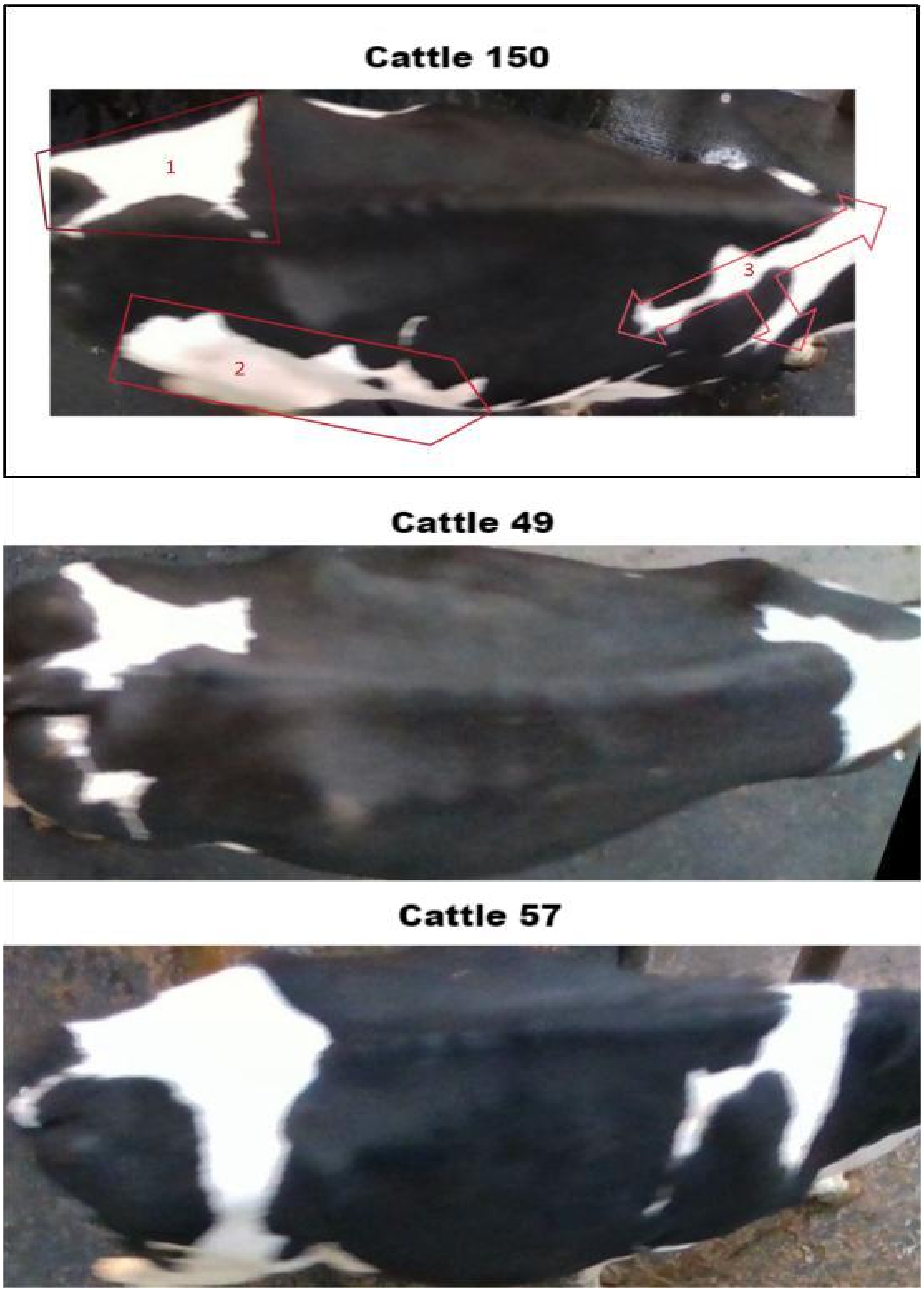


Figure 6 Cattle 150/49/57 Comparison

For example, we can designate the three white blocks on cattle 150 as a trapezoid, a Y, and a pentagon, which correspond to 1, 2, and 3. While Cattle 49 could be seen as a combination of 1 with a smaller 2 and a reversed 3, the cattle 57 is a combination of 1 (above the spine) and 3.

Though our human could distinguish between the small differences, the model might struggle with integrate those elements as a unique identifier for each individual cow, and having trouble distinguish subtle form orientation differences.

4.3.3 Conclusion on Misclassification

Our analysis of the cattle identification model shows the strengths and areas for improvement. The model shows a high overall accuracy, with only 15 misclassifications out of 607 test images. However, these errors indicate there are still many challenges in fine-grained visual classification tasks:

1. Pattern Similarity: The model is struggling to differentiate cattle with similar black and white patterns, especially those with large black areas and similar white patch distributions.

2. Feature Focus: In some cases, the model is over-relying on specific local features (e.g., teardrop-shaped marks near the spine) rather than other contexts and overall patterns.

3. Edge Pattern Complexity: Irregular edge patterns between black and white areas is a challenge for our model, mainly when local similarities exist within different patterns.

4. Modular Pattern Recognition and Composition: In several cases, the model could recognize the pattern in a modular way and re-composition them into a complete picture; this caused the model to misclassify cattle that have more than one similar feature, although the location or the directions of those patterns are not identical.

5 Conclusions and Future Work

5.1 Findings and Break Though

This dissertation has explored the development of an automated cattle identification system based on their unique color block patterns using self-supervised learning by employing the ResNet50 architecture. The model (with our adjustment) achieved 98.2% in test accuracy and 97.3% in transfer learning accuracy after running 527 epochs.

The aim of our research is to improve the efficiency and animal welfare in cattle management. After analysis the result of our work, we have below findings and break through which could greatly answer our research questions:

5.1.1 Primary Findings:

1. Model Performance: above three preselect and modified models, ResNet-50 is the most effective model for cattle identification. It outperformed the custom models and EfficientNetB0 in the accuracy of validation, test, and transfer learning scenarios and shows superior stability than EfficientNetB0 within a simple hyper-parameter. Highlighting its robustness in handling the variability in cattle color patterns.

2. Impact of Image Preprocessing: The experiments are held between the original images, the grayscale images, and the resolution-reduced image. Through our experiment, we found that although the color information is crucial to maintaining a high identification accuracy, the grayscale images keep sufficient distinguishing features. Conversely, reducing the image resolution would significantly affect model performance, suggesting that resolution plays a much more critical role in capturing intricate pattern details essential for identification than color in the cattle identification problem.

3. Challenges in Misclassification: Although we achieved a high transfer learning accuracy, the model is having trouble in distinguishing cattle with complex or similar patterns. It shows an over-reliance on local features and challenges in dealing with complex edge patterns. Also, the structure of ResNet50 has caused a modular pattern recognition problem. These cases point to the need for model refinement that can better overcome this kind of fine-grained visual differences.

5.1.2 Break Through

1. Feature selection: Compared to the previous studies, which used cattle muzzle print, iris, and body movement and behaviour as identification features, this study focuses on using the unique color block on the back of cattle for recognition training. This distinctive feature occupies a large area on the cattle body, which is much easier to obtain and not easily blocked. Using this feature, we can avoid body damage such as ear tags or chip injections and improve animal welfare.

2. Computational Efficiency: Compared to methods requiring high-resolution images like muzzle print or retinal image, this study selected a larger pattern which could reduce computational demands and learning speed, we also downscaled the original cattle images from 1280x720 pixels into 64x64 pixels while maintaining a high identification accuracy.

5.2 Limitations

Although our model currently has a high accuracy rate in identifying individuals in known herds, due to below limitations, the model is still having room for improvement:

1. Data Constraints: The dataset we used, although it already contained 181 cows with nearly 8,000 images, the fact that it contained many cows with very similar characteristics presented challenges to model learning and differentiation. The size of the sample and the quality of the sample of individuals may have contributed to specific misclassification problems. In addition, after data segmentation, the number of transfer learning datasets available becomes too small that the classification metrics can be misleading and may not accurately reflect the model's prediction capability.

2. Computational Resources: The need for high computing power, especially when adjusting the parameters of advanced deep learning models, greatly limits the number of debugging we can do. In practice, each time a new cattle model is introduced, it should be partially or retrained to increase the robustness of the identification library, which may cause some distress in resource-constrained farm environments.

3. Model Complexity: EfficientNetB0, while offering high accuracy, requires careful tuning that its parameters needed to be carefully adjusted to avoid serious gradient explosion; we did not conduct an in-depth exploration of this model due to various limitations. Thus, we did not know whether it had the potential to surpass our ResNet50 after adjustment.

4. Limited Model Integration: The current study focused solely on exploring the ResNet-50 architecture. From our literature review, there is potential for further improvement by integrating ResNet50 with other model types further enhance the model's ability and accuracy. Such as the triple stream networks, which using the distance between features to identify individuals.

5.3 Suggestions for Future Research

To enhance the applicability and robustness of this cattle identification systems for practical use, future research should consider the following directions:

Dataset Expansion and Diversity: Expand the dataset, including numbers of cattle and the diversity of images from different angel particularly for individuals with challenging pattern similarities. This can help improve model training and accuracy. Incorporating more environmental variability will also aid in creating a model with better capability.

Adaptive Learning Techniques: The inclusion of adaptive learning techniques enables the model to constantly learn and update from new data without the need for comprehensive retraining. This can be particularly beneficial in farm environments where new cattle are introduced regularly, addressing computational resource constraints in practical applications.

Multi-feature Methods: Investigate the potential of combining color block patterns with other non-invasive identification methods, such as artificial marking or behavioral patterns. This multi-feature method could enhance the identification system's overall accuracy and reliability while maintaining harmless to cattle.

Model Integration and Enhancement: Explore the integration of ResNet50 with other model architectures, such as triple-stream networks or Siamese networks. These models is excellent in identify feature distances, which could significantly improve the system's ability to distinguish between cattle with similar patterns. This integration could potentially overcome the current limitations in differentiating subtle differences between similar cattle.

Ethical and Welfare Considerations: Further explore the ethical implications and animal welfare benefits of this non-invasive identification method. Studies comparing stress levels and overall well-being of cattle identified through this method versus traditional methods could provide valuable insights.

Cross-breed Applicability: Extend the research to include multi breeds of cattle to test and enhance the model's generalization capabilities. Explore the pattern difference between breeds would increase the system's applicability across different types of cattle farms globally.

Interpretability and Explainability: Develop methods to increase the interpretability of the model's decision-making process. This could involve developing techniques like gradient-weighted class activation mapping (Grad-CAM) to visualize which parts of the image are most influential in the model's decisions process, providing insights into how the model identifies individual cattle.

Integration with Other Management Systems: Explore ways to integrate this identification system with broader farm management software, potentially linking individual cattle identification with health records, milk production data, infrared thermal image and other relevant information for comprehensive livestock management.

By pursuing these research directions, future work can address the current limitations of the system and potentially revolutionize cattle management practices, offering more efficient, ethical, and accurate methods of livestock identification and monitoring.

References

[1] Senbagam, B. and Bharathi, S. (2024) ‘Animal detection in wildlife conservation using Deep Learning’, ICT: Cyber Security and Applications, pp. 217–228. doi:10.1007/978-981-97-0744-7_18.

[2] Kavitha, D.R. et al. (2023) ‘Wild animal detection using CNN’, International Journal for Research in Applied Science and Engineering Technology, 11(5), pp. 678–683. doi:10.22214/ijraset.2023.51500.

[3] Gunaratnam, A. et al. (2024) ‘Computer vision in livestock management and production’, Engineering Applications in Livestock Production, pp. 93–128. doi:10.1016/b978-0-323-98385-3.00002-5.

[4] Wang, J. et al. (2023) ‘Op mask R-CNN: An advanced mask R-CNN network for cattle individual recognition on large farms’, 2023 International Conference on Networking and Network Applications (NaNA) [Preprint]. doi:10.1109/nana60121.2023.00104.

[5] He, K. et al. (2016) ‘Deep residual learning for image recognition’, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) [Preprint]. doi:10.1109/cvpr.2016.90.

[6] Collazos Huertas, D.F., Gómez Gómez, G.S. and Álvarez Meza, A.M. (2021) ‘Image-based animal recognition based on Transfer learning’, Scientia et Technica, 26(03), pp. 406–411. doi:10.22517/23447214.24538.

[7] Zhou, Z. et al. (2023) ‘Improving the classification accuracy of fishes and invertebrates using residual convolutional neural networks’, ICES Journal of Marine Science, 80(5), pp. 1256–1266. doi:10.1093/icesjms/fsad041.

[8] Mendu, A., Sehgal, B., & Mendu, V. (2022). Cattle Detection Occlusion

Problem. https://doi.org/10.48550/arXiv.2212.11418

[9] Zhang, R. et al. (2023) ‘A cascaded individual cow identification method based on Deepotsu and EfficientNet’, Agriculture, 13(2), p. 279. doi:10.3390/agriculture13020279.

[10] Yin, X. et al. (2020) ‘Using an efficientnet-LSTM for the recognition of single cow’s motion behaviours in a complicated environment’, Computers and Electronics in Agriculture, 177, p. 105707. doi:10.1016/j.compag.2020.105707.

[11] Yang, Y. et al. (2022) ‘Real-time cattle interaction recognition via Triple-Stream Network’, 2022 21st IEEE International Conference on Machine Learning and Applications (ICMLA) [Preprint]. doi:10.1109/icmla55696.2022.00016.

[12] Agarwal, D. (2023) ‘Segmentation of cattle using color-based skin detection approach’, Soft Computing and Signal Processing, pp. 621–633. doi:10.1007/978-981-19-8669-7_55.

[13] Kaur, A., Kumar, M. and Jindal, M.K. (2023) ‘Cattle identification from muzzle print image pattern using hybrid feature descriptors and SVM’, Springer Proceedings in Mathematics & Statistics, pp. 479–489. doi:10.1007/978-3-031-15175-0_39.

[14] Allen, A. et al. (2008) ‘Evaluation of retinal imaging technology for the biometric identification of bovine animals in Northern Ireland’, Livestock Science, 116(1–3), pp. 42–52. doi:10.1016/j.livsci.2007.08.018.

[15] Liu, Liping., Gao, Yunqi., Shao, Gang., Jia, Zhengquan., Liu, Zheng., Wang, Jiahui., Liu, Tao. (2019). A tracking four-axis system realization method based on color block identification and an aircraft using the method. https://typeset.io/papers/a-tracking-four-axis-system-realization-method-based-on-pqpo9eth74

[16] Wang, J. et al. (2023) ‘Open pose mask R-CNN network for individual cattle

recognition', IEEE Access, 11, pp. 113752–113768. doi:10.1109/access.2023.3321152.

[17] Grosz, M. D., & MacNeil, M. D. (1999). The "spotted" locus maps to bovine chromosome 6 in a Hereford-Cross population. The Journal of heredity, 90(1), 233–236. https://doi.org/10.1093/jhered/90.1.233

[18] Gao, J., Burghardt, T., Andrew, W., Dowsey, A. W., & Campbell, N. W. (2021). Towards Self-Supervision for Video Identification of Individual Holstein-Friesian Cattle: The Cows2021 Dataset. ArXiv. /abs/2105.01938

[19] Kumar, S. et al. (2017) 'Analytical Study of Animal Biometrics: A technical survey', Animal Biometrics, pp. 21–78. doi:10.1007/978-981-10-7956-6_2.

[20] Allflex. (n.d.). Livestock Management. Retrieved August 8, 2024, from https://www.allflex.global/uk/livestock-management/

[21] Gobbo Oliveira Erünlü, N., Rieder, S., Kuntzer, T., Bérard, J., & Wellnitz, O. (2023). Effects of sensor ear tags with twin pin fixing system on health and well-being of cattle. Auswirkungen von Sensorohrmarken mit Doppelstift-Befestigungssystem auf die Gesundheit und das Wohlbefinden von Rindern. Schweizer Archiv fur Tierheilkunde, 165(7), 512–523. https://doi.org/10.17236/sat00399

[22] Kandemir, Ç. et al. (2023) 'Performance evaluation of leg and ear numbers in radio frequency identification systems (RFID) in sensitive livestock products in goat breeding', Ciência Rural, 53(6). doi:10.1590/0103-8478cr20210801.

A Appendix

A.1 Hardware and Software

A.1.1 Hardware Information

CPU : Intel Core i7-13700KF, 16 Core, 3400.0 MHz

GPU Name: NVIDIA GeForce RTX 4070 Ti, 12GB

Memory: 32GB

GPU Driver Version: 552.22

A.1.2 Software Information

Operating System: Windows 10

OS Version: 10.0.19041

Coding: Vs code

Python Version: 3.8.19 (default, Mar 20 2024, 19:55:45) [MSC v.1916 64 bit (AMD64)]

A.2 Installed Packages

scikit-learn==1.3.2

tensorflow==2.9.0

pandas==2.0.3

numpy==1.24.3

matplotlib==3.7.5

gputil==1.4.0

platformdirs==4.2.0

psutil==5.9.8

opencv-python==4.10.0.84