Livestock Identification using Deep Learning

Kevin Sebineza
Carnegie Mellon University
ksebineza@andrew.cmu.edu

Schadrack Niyibizi
Carnegie Mellon University
sniyibiz@andrew.cmu.edu

Chukwuemeka Livinus Nkama
Carnegie Mellon University
lnkama@andrew.cmu.edu

***Abstract*—Livestock identification is crucial for various processes, such as vaccination, insurance, and disease monitoring. However, traditional methods such as ear tagging and tattooing proved to be inefficient and harmful to the cow. In this study, we utilized advanced deep-learning frameworks to identify cattle based on muzzle print imagery. Our approaches involved a ResNet50 architecture integrated with a dot product matching algorithm and a Siamese network built upon a convolutional neural network backbone. Both models demonstrated high levels of training and validation accuracy and exhibited significant scalability. The Siamese network configuration achieved superior testing accuracy, reaching over 93.95%. Meanwhile, the ResNet50 model attained a testing accuracy of 84.1%. These results highlight the potential of deep learning frameworks in enhancing animal identification systems, offering a more humane and effective alternative to traditional methods.**

***Index Terms*—Deep Learning, ResNet50, Siamese Network, Dot Product**

Google Drive
GitHub Repository

I. Introduction

Animal biometrics, an emerging field of research at the intersection of computer vision, pattern recognition, and cognitive sciences, aims to identify animals based on their physiological and phenotypic properties [1]. This field has significant implications for livestock identification, where animal identification and traceability are crucial not only for health management but also for managing food production.

Traditional animal identification methods can be categorized into mechanical-based identification, electrical signal-based identification, and biometric feature-based identification [1]. While these methods have their merits, they also come with several challenges. For instance, ear-tagging, which involves the use of metal clips and plastic tags, can lead to infections in animals. Moreover, these methods can be uncomfortable for the animals and the identification tags can become lost, stolen, or damaged. There is also the risk of fraudulent insurance claims intended to deceive the insurance provider [1]. These challenges underscore the need for more effective and reliable biometric systems.

In recent years, deep learning systems have been developed to address these challenges. Models such as the convolutional neural network, deep belief network, and recurrent neural network have demonstrated a high degree of accuracy [2]. However, these methods lack scalability as the addition of a new cattle to the database necessitates retraining the model.

In response to this, we propose a novel cattle recognition system that leverages a Siamese network and a dot product approach between feature vectors. This approach achieves scalability without the need for re-training the model. The key contributions of this study are as follows:

1) A novel feature extraction method that uses a dot product approach to predict the identity of a given cattle in a scalable manner, offering a significant improvement over pre-existing methods.
2) A demonstration of the viability of the Siamese Network as a means to identify individual cattle without the need for retraining.
3) Data augmentation of the Beef Cattle Muzzle Noseprint dataset [3].

II. Literature Review

A. Introduction to Biometric Systems

1) History of Biometrics: Biometrics is a science that deals with the verification and identification of an entity based on behavioural and physiological traits [2]. Biometric systems have been developed to recognize humans and they work by leveraging physiological features such as the retina, gait, face, iris, voice and finger veins [4]. For a feature to qualify as a biometric trait, it should satisfy the following measurements: universality, distinctiveness, invariance, collectability, performance, acceptability, simplicity, circumvention and cost efficiency [5]. That aside, biometric systems are typically automated; however, this has not always been the case [6].

Although there are a variety of ways to identify an individual, fingerprint recognition is the oldest form of biometric recognition and goes back to about 6000 B.C [6]. For example, in ancient babylon, fingerprints were used on clay tablets for business transactions and the chinese were known to stamp palm prints on paper with ink as a means of identifying children [6]. Johannes Evangelista Purkinje performed the first modern study and proposed a system of fingerprint classification in 1823; since that time, fingerprint recognition has undergone a lot of evolution and other biometric techniques have been discovered [7].

2) Overview of Biometric Technologies: Biometric systems, irrespective of the species involved, typically comprise four main components: image acquisition, feature extraction, matching, and database templates [2]. The process begins with image acquisition, where the subject's image is captured for the analysis of its morphological biometric features [1]. If the image quality is subpar, it undergoes pre-processing to facilitate the extraction of appropriate features [2]. Following pre-processing, the feature of interest is extracted. The extracted feature is then matched with the registered image (an image from the database) to obtain a matching score [2]. Finally, this score is compared to a threshold to decide whether to verify or reject the subject's identity [2].

Biometric systems can operate in one of two modes: identification or authentication (verification) [6]. In authentication mode, the system aims to confirm the individual's claimed identity [6]. This is achieved by performing a 1-to-1 match between a biometric template stored in the database and the query template [2]. Examples of such systems include computer logins, access control, ATMs, and e-commerce [2]. In identification mode, the user's biometric template is compared with all the templates in the database (a 1-to-N comparison) [2]. Identification applications include passports, ID cards, and driving licenses [5].

When developing a biometric system, it's crucial to consider factors that affect its performance [2]. These factors include image quality, the biometric algorithm used, and environmental conditions surrounding the system [5]. The accuracy of a biometric system is typically measured in terms of sample acquisition error and performance error [2]. Sample acquisition error, often due to environmental factors, occurs when an individual fails to enroll due to the sample image being noisy [2]. Performance error, on the other hand, measures the accuracy of the biometric system when deployed in real-time [2]. Metrics used to measure performance error include the false rejection rate, false acceptance rate, ROC curve, equal error rate, identification rate, false-negative identification rate, false-positive identification error rate, and the cumulative match characteristic curve [2].

B. Deep Learning in Biometric Systems

Deep learning has been extensively researched in various biometric systems in recent years, with models employed ranging from Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) to Restricted Boltzmann Networks, Deep Belief Networks, and Stacked Auto-encoders [2].

Among these, CNNs have shown exceptional performance with various biometric traits. Comprising convolutional, pooling, and fully connected layers, each neuron in these layers extracts a pattern in a local region of the subject image [8]. CNNs are considered more computationally effective than other networks due to their fewer parameters [2]. Some notable CNN architectures include ResNet, AlexNet, DensNet, GAN, and GoogleNet [9]–[13].

RNNs, being sequential-based networks, are used for their robustness to noise during the training process [2]. Restricted Boltzmann Machines, also known as unidirectional graphical models, are utilized by Deep Belief Networks, which are unsupervised in nature [2], [14]. However, they are challenging to optimize and computationally expensive to train due to their greedy nature [2].

Stacked auto-encoders, typically used for unsupervised learning, consist of an encoder and a decoder. The encoder learns representative information, while the decoder reconstructs it [2]. These auto-encoders have proven useful in biometric systems.

The integration of deep learning in biometric systems has significantly improved image quality and system performance [2]. Other intriguing networks, such as the Siamese network, are also used in biometric systems. Siamese networks allow for image recognition without the need for a pre-constructed database. They are also scalable, hence eliminating the need for retraining, unlike most deep learning approaches [15].

C. Animal Biometrics

1) Exploring Biometric Systems for Animals: While biometric systems are typically designed for humans, they can also be applied to animals [16]. The identification of individual animals is crucial for various reasons, including the verification of animal ownership, bio-security, and animal tracking for research and agricultural purposes [17].

Traditional methods for identifying animals involve body measurements and marking individual animals. These methods include ear-tagging, ear tattoos, freeze branding, hot iron branding, embedding microchips, ID-collars, and paint and dye-based methods [17]. Despite their widespread use, these methods have several challenges: they can easily get lost, damaged, or duplicated, and they can pose health risks to animals [18]. To address these issues, researchers have been developing animal identification systems using computer vision and image processing techniques [19].

Current approaches for animal identification involve extracting salient features from the biometric characteristics of animals [20]. These features include joint stripe patterns, skin color patterns, face images, muzzle prints, iris-based patterns, and retinal vascular patterns [1], [20].

A typical animal biometrics system comprises six key components: sensors for data acquisition, feature extraction, storage, similarity matching of the query animal with stored templates in the database, decision-making based on matching scores and defined threshold values, and utilization of extracted feature characteristics [1]. Current state-of-the-art animal biometric systems include SLOOP and the ECOCEAN whale shark identification system [1].

2) The Importance of Cattle Identification: The process of cattle registration and identification plays a crucial role in preventing the manipulation, verification, and identification of swapped cattle [1]. This includes the implementation of recognition systems that raise alerts for fraudulent insurance claims related to registered cattle [1].

International bodies have recognized the need for the design of framework-based systems that incorporate the Internet of Things (IoT) for the recognition and tracing of animals [7], [16]. The benefits of such systems are manifold: they help control the spread of critical diseases among animals, reduce governmental costs, and minimize the loss of livestock [1].

D. Cattle Identification Techniques

1) Cattle Identification Using Facial Images: [1] proposed an approach for identifying cattle using facial images. This method employs Principal Component Analysis (PCA) to derive eigenvectors that abstract the features of cattle from an image. Initially, the images representing the cattle are pre-processed and enhanced using contrast-limited adaptive histogram equalization. Subsequently, the covariance matrix is used to generate discriminatory eigenvectors, which are sorted based on their importance. During the identification phase, the eigen-face feature vectors are matched with stored feature templates, resulting in an identity verification for the subject image [1].

2) Cattle Identification Using Muzzle Prints and Image Processing Techniques: [21] noted that the recognition of minutiae points in muzzle images bears similarity to fingerprint recognition. The artifacts of these points can be categorized into two attributes: beads and ridges. Beads, akin to islands, are separated from each other by ridges. These attributes contain texture information that forms the basis for recognizing cattle [1]. The identification process for muzzle prints involves several steps: preprocessing, segmentation of muzzle point images, extraction of bead and ridge features from the muzzle point images using local texture descriptor-based techniques, appearance-based feature extraction, and the use of a chi-square distance-based matching technique to compute the dissimilarity scores between the stored muzzle print and the test muzzle point images [1].

3) Cattle Identification Using Muzzle Prints and Deep Learning: While appearance-based feature extraction and representation algorithms can yield satisfactory results, they may struggle to recognize cattle when the quality of test images is low [1]. These approaches include Local Binary Pattern (LBP) [22], Circular-LBP [23], Scale-Invariant Feature Transform (SIFT) [24], Dense-SIFT [24], Speeded Up Robust Feature (SURF) [25], and Vector of Locally Aggregated Descriptor (VLAD) techniques. In cases where these methods fail, deep learning can provide a solution [1].

To utilize deep learning, muzzle print images must first be pre-processed into grayscale images to reduce artifacts and noise inherent in the images [1]. Several deep learning architectures can be used to extract features, including but not limited to: Convolutional Neural Networks (CNNs), Deep Belief Networks, and Auto-encoders [1]. CNNs consist of a stack of convolutional and pooling layers, culminating in a final softmax layer to perform the classification process [1]. Deep Belief Networks are graphical deep learning models that use Restricted Boltzmann Machines (RBMs) during training to extract necessary features [26]. They are used because they employ unsupervised learning for the feature extraction and representation of the muzzle point image [1]. Stacked Denoising Auto-encoders are also used on muzzle print images; they encode and decode the extracted texture features for better feature representation in the feature space [1].

E. Prospects and Challenges in Animal Biometrics

Despite significant advancements in the development of biometric systems, challenges remain. [1] proposed several strategies that could further advance the field of animal biometrics. These include the design of multimodal fusion-based animal biometric recognition systems to enhance accuracy, the use of unmanned aerial vehicles (UAVs) to develop ecological databases for expanded research, and the application of reinforcement learning to enable biometric systems to better adapt to constrained environments.

III. Methodology

This project aims to build a scalable AI-powered model that can identify an individual cow using its muzzle print image within 30 seconds. The proposed pipeline involves training deep learning architectures on a representative dataset and ensuring the scalability of the model, where model retraining is not required once a new cow is introduced.

A. Dataset

1) Description of the dataset: The ”Beef Cattle Muzzle/Noseprint Database” [3]. is a collection of 4,923 photographs of the muzzles of 268 feedyard yearlings. These photos were taken in the Midwest United States between March and July 2021, using a mirrorless camera with a 70-300 mm lens. The breeds primarily focused on were Angus, Angus x Hereford, and Continental x British crossbreeds. The dataset is meant for individual cow recognition and is organized into folders for each animal, with an average of 12 high-resolution images per subject. The images are carefully cropped only to show the muzzle area and have a maximum resolution of 24 MB.

Fig. 1. Sample images from the Beef Cattle / Nose Print Dataset

Although the "Beef Cattle Muzzle/Noseprint Database" is useful for identifying cattle, its high-quality imagery is not representative of real-world situations, where images captured on smartphones often have lower quality. We used different image processing techniques to create a model that can handle farm-captured images with varying quality. As a result, we developed an augmented dataset that combines the original high-resolution images with artificially degraded versions to simulate lower-quality images that are more typical of on-field conditions.

2) Data Preprocessing: This study utilized advanced image transformation techniques to replicate low-quality images typically captured by smartphones. The process involved modifying the image resolution, illumination, and spatial orientation. For resolution and lighting adjustments, the openCV library was used, which included brightness and contrast modulation, along with the introduction of Gaussian noise. For spatial orientation adjustments, the torchvision transformations module was used, which implemented random perspective changes, horizontal flipping, and arbitrary rotation.

Each original image underwent these transformations, resulting in a secondary dataset with an average of 48 altered images per cow, simulating lower-quality visual data. The transformed dataset was then combined with the initial high-quality dataset, creating a comprehensive collection of 12,864 images with both high and low-quality representations of the same 268 cows. Images depicting the same cow were treated as inputs, and the cow's unique identifier served as the target label for classification purposes.

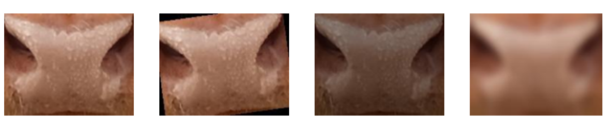

Fig. 2. Illustration of the original image and applied transformations

After the image transformation phase, the dataset was divided into four subsets: the training set, validation set, feature extraction subset, and testing set. The training and validation subsets were used for model training and hyperparameter optimization, while the feature extraction subset generated representative feature vectors. Lastly, the testing set was utilized to evaluate the model's ability to recognize new cows introduced to the dataset.

B. Pipeline Description

The modeling phase had two main goals: to create an image identification model that is accurate and scalable. The accuracy component aimed to achieve high performance during training and validation while maintaining robust accuracy when the model encounters images of new cows. The scalability aspect aimed to enable the model to recognize pictures of newly introduced cows without requiring a complete retraining process. To accomplish these goals, we developed two distinct operational pipelines: one for identifying cows based on their images and another for incorporating images of new cows without full model retraining.

1) Prediction Pipeline: In this process, an incoming image goes through a feature extraction mechanism of a trained architecture. This step generates a set of feature representations or embeddings for the image. A matching algorithm then compares these embeddings with the existing feature vectors for each cow in the database. The cow whose feature vectors most closely align with the input image's embeddings is then identified as the most probable match.

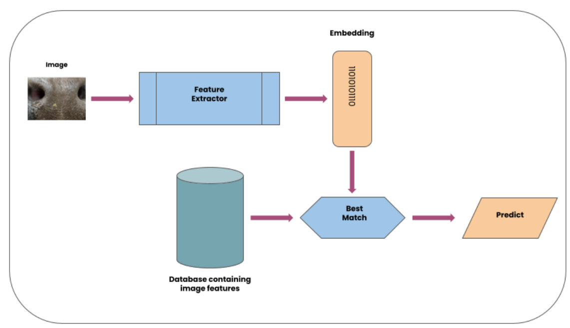

Fig. 3. Flow chart showing how we predict a cow identity given an image.

2) Scalability Pipeline: When introducing a new cow into the system, at least five images showcasing the cow's unique muzzle print are collected. These images are then fed into the trained feature extraction model, which generates embeddings for each image. These newly produced feature vectors are subsequently aggregated to form a singular representative feature vector that encapsulates the essence of the new cow's muzzle print. This consolidated feature vector is then stored in the database. This methodology allows for the seamless addition of new cows into the system, avoiding the need for a comprehensive retraining of the model with the entire image dataset.

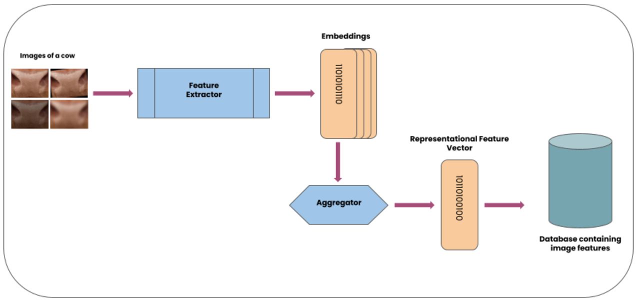

Fig. 4. Flow chart showing how the model works by adding images of a new cow without retraining.

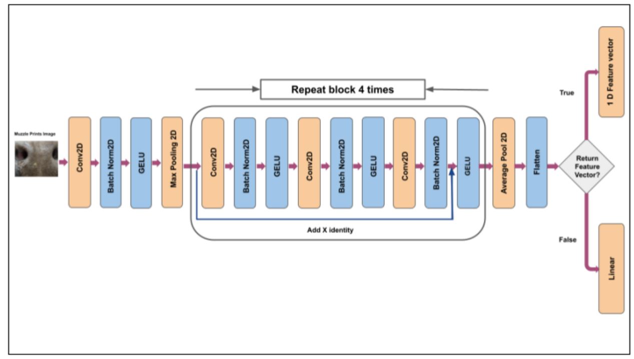

Fig. 5. Chart showing the ResNet architecture with the condition of passing the flattened feature vectors to the Linear classifier during the training or returning the embeddings during testing

C. Model Selection

Two modeling approaches were applied, simulating the pipelines highlighted in fig 3 and fig 4, namely, the Siamese Network and the Resnet50 with the Dot Product. Both architectures focused on training a feature extractor and applying effective best-match and aggregator functions. The Siamese network was selected due to its application in human biometrics, while the ResNet50 architecture was reimplemented due to yielding great accuracy from the previous work, as highlighted in the literature review.

1) ResNet50 with Dot Product: We implemented the ResNet architecture from scratch to learn feature representations to use for image identification. We defined the block class to define the fundamental building block of the ResNet architecture for capturing and transforming image features. Each block comprises three convolutional layers, each followed by batch normalization and the GELU (Gaussian Error Linear Unit) activation function. The choice of using GELU instead of the popular ReLU stems from its smoothness and differentiation ability.

We also defined the Resnet class which assembles the blocks into a complete neural network. The ResNet class starts with a 7x7 convolutional layer to initially process the input image. This is followed by batch normalization and a GELU activation function. To reduce the image size, a max-pooling layer is used. The core of the model comprises four primary layers (layer 1 to layer 4), constructed using the Block class. Following the four layers, an adaptive average pooling layer generates a fixed-size feature map, which is then flattened into a 1D vector. Optionally, if the return_features argument is set to True, the model can return these features, allowing us to make the model scalable by not training the model everytime another image class is introduced. Lastly, a fully connected layer (fc) is used to map these features to the desired number of output classes for classification. This architecture's flexibility and adaptability make it well-suited for diverse image classification tasks.

Training Methodology

The training approach for the ResNet50 model aimed to optimize parameters using the available image and label data from both the training and validation datasets. A batch of 32 images, along with their corresponding labels, was fed into the complete ResNet50 architecture, which included a linear classification layer. This helped to calculate the cross-entropy loss and perform backpropagation. To optimize the training process, an Adam optimizer was used, and its learning rate was adjusted based on a scheduler with a threshold of 0.05. The comprehensive model included more than 35 million trainable parameters.

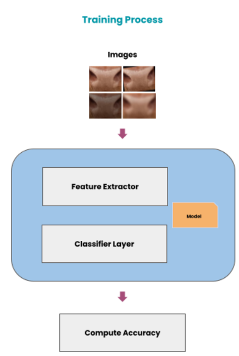

Fig. 6. Chart showing the training and validation loop for the ResNet50 architecture

Prediction Process

After completing the training cycle, images from the feature selection were passed through the ResNet50 architecture, excluding its final classifier layer. This step generated individual feature vectors for each image. These vectors were combined into a single representative vector using the KMeans clustering algorithm and stored in a dictionary-style database. To evaluate the model's effectiveness on new images, the testing set images were fed into the trained ResNet50 model, which acted as a feature extractor, producing a feature vector denoted as x_i. This vector was then compared with the stored vectors (x) in the database using the dot product. The feature vector in the database that produced the highest dot product score in relation to x_i was used to identify the cow associated with the input image.

$$\text{Dot product} = x_i^T x \tag{1}$$

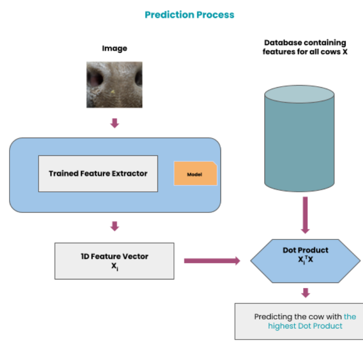

Fig. 7. Chart showing the process of predicting the cow the image belongs to using the Dot Product as the best-match function.

2) Siamese Network: The Siamese network learns how to minimize "some" distance between similar features and maximize the distance between dissimilar features. The network is able to achieve this by using identical sub-networks with similar parameters. For our case we re-implemented the siamese network base architecture by using a convolution network as the feature extractor and the contrastive loss as the aggregator function that evaluates the similarity between two image feature vectors.

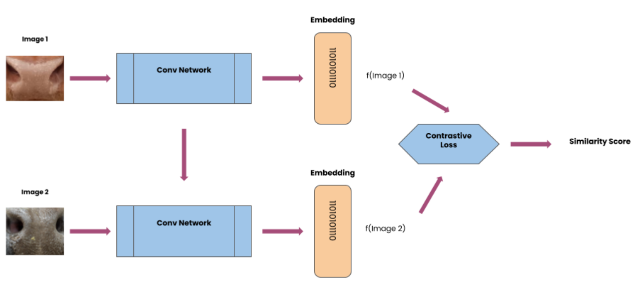

Fig. 8. Chart showing the Siamese network architecture

For the feature extractor, in this case, the convolution network, it takes an input shape, flattens it, and passes it through two dense layers with ReLU activation and dropout for regularization. Finally, it applies L2 normalization on the outputs, which helps to keep the learned features at a consistent scale. As observed from fig 8, the two arrows between the two convolution networks demonstrate the similarity nature of these networks, as they share the same parameters. The contrastive loss which measures the similarity between two vectors is a distance-based loss function that uses the Euclidean distance to determine the similarity and dissimilarity between two feature vectors.

$$\text{Distance} = \sqrt{[G_w(x_1) - G_w(x_2)]^2} \tag{2}$$

$$\text{Contrastive Loss} = (1-Y)\tfrac{1}{2}(\text{Distance})^2 + Y\tfrac{1}{2}[\max(0, m-\text{Distance})]^2 \tag{3}$$

Training Process

To develop the convolutional neural network, the training and validation datasets were utilized to fine-tune the network. The network's convolutional architecture processed batches of 32 images, along with their corresponding labels. This led to the creation of feature vectors. The RMSprop optimizer was utilized to adapt the network's parameters by minimizing the contrastive loss. A constant learning rate of 0.001 was used for this optimization.

Prediction Process

The Siamese network performs pairwise comparisons between the feature vector generated from the input image and the feature vectors already present in the system to identify the specific cow based on its muzzle print image. During this process, the network calculates the contrastive loss for each pair. When the similarity score between the input image's feature vector and a stored vector falls within a predefined threshold, the input image is then assigned the label corresponding to that of the matched pair.

IV. Results and Discussion

The two model architectures, ResNet50 with the dot product and the Siamese Network were trained and tested on the augmented dataset containing high-quality and low-quality muzzle print images.

A. ResNet50 with the Dot Product

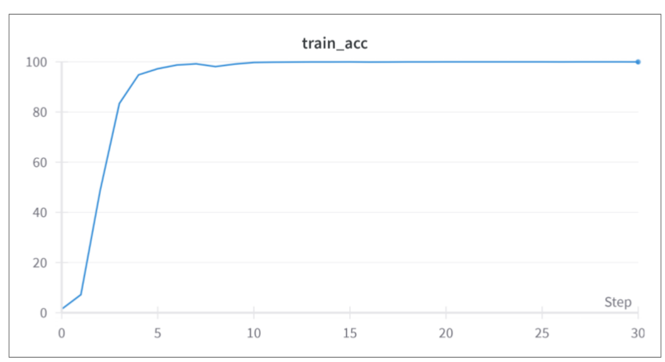

Fig. 9. Chart showing the training accuracy over epochs

1) Training and Validation Accuracy: During the training process, the ResNet architecture quickly converged and achieved the training accuracy of 99.98% surpassing the accuracy achieved by previous studies that used the same architecture on the dataset. The architecture also achieved an impressive validation accuracy of 98.27%.

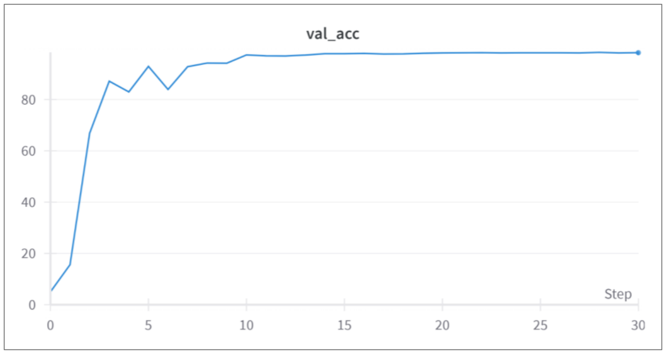

Fig. 10. Chart showing the validation accuracy over epochs

2) Testing Accuracy on Newly Added Cows: We experimented the ability of the architecture to predict on newly added cows, aiming to evaluate the reliability of the model the more new cows are introduced.

Dot Product Test Accuracy Table When Adding new cows

Trained on → / Adding new cows ↓	25	50	100	180
50	8.8% (new 25)	54.0% (new 0)	N/A	N/A
100	6.6% (new 75)	43.6% (new 50)	59.6% (new 0)	N/A
125	7.2% (new 100)	40.2% (new 75)	58.4% (new 25)	N/A
150	8.0% (new 125)	38.1% (new 100)	54.4% (new 50)	N/A
180	9.3% (new 135)	35.7% (new 130)	54.9% (new 80)	84.1% (new 0)
200	8.7% (new 175)	33.2% (new 150)	51.9% (new 100)	83.0% (new 20)
210	8.9% (new 185)	32.8% (new 160)	52.2% (new 110)	82.95% (new 30)
220	9.1% (new 195)	32.3% (new 170)	52.5% (new 120)	82.5% (new 40)
230	8.2% (new 205)	30.0% (new 180)	50.5% (new 130)	77.1% (new 50)
250	7.6% (new 225)	29.4% (new 200)	50.1% (new 150)	78.2% (new 70)
268	2.3% (new 243)	28.8% (new 218)	48.9% (new 168)	77.3% (new 88)

Fig. 11. Chart showing the ResNet50 with the dot product performance on newly added cows

The table on fig 11 shows different test accuracy given the number of trained cows and newly added cows. The first row of the data presentation displays different groups of trained cows, each having an average of 48 images. On the other hand, the first column indicates the count of new cows introduced to the system. For instance, when the model was initially trained with 25 cow classes, the addition of 25 new cows resulted in an accuracy of 8.8%. Conversely, training the system with 50 cow classes produced an accuracy of 54%, which dropped to 43.6% when 50 new cows were added. Finally, training the system with 180 cow classes resulted in a test accuracy of 84.1%, which slightly decreased to 83% when 20 new cows were added. Generally, accuracy increased as more training images were used but decreased when more new cows were introduced into the model.

Furthermore, the dot product method took an average prediction time of 1 second to identify a cow from an image, which met the project goal of achieving predictions within 30 seconds.

3) Performance on the Collected Field Images: Our team ventured to a local livestock farm located in Kigali, Rwanda, collecting an average of 5 images from 6 cows to test on the model's performance.

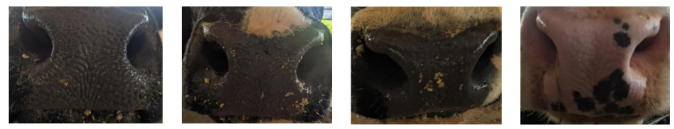

Fig. 12. Sample images of cows collected from a local livestock farm

The ResNet50 with the Dot Product architecture achieved 100% testing accuracy on the newly added 6 cows, leading to a conclusion that this approach heavily relies on the images used to produce representative feature vectors, since the stored embeddings are the ones used in the calculating the dot product with the input image.

B. Siamese Network

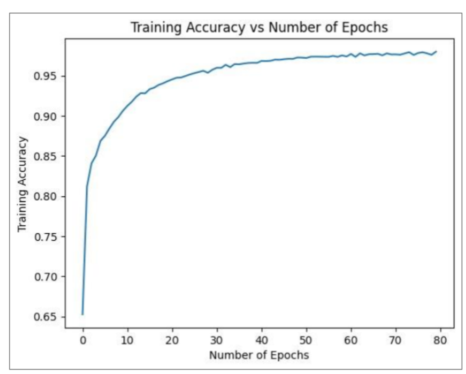

Fig. 13. Chart showing the training accuracy over epochs

1) Training and Validation Accuracy : Contrary to the ResNet50 architecture, the convolution network within the siamese network did not converge quickly, but it also performed well with a training accuracy of 98%. The architecture also achieved a validation accuracy of 95%.

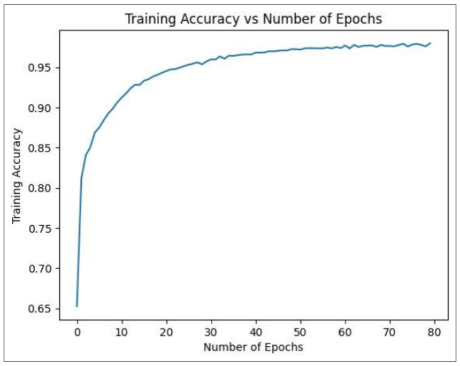

Fig. 14. Chart showing the validation accuracy over epochs

2) Testing Accuracy on Newly Added Cows: The Siamese Network was tested similar to the ResNet50 architecture producing impeccable results when new cows were introduced to the model.

SIAMESE Test Accuracy Table When Adding new cows

Trained on → / Adding new cows ↓	25	50	100	180
50	73% (new 25)	96.7% (new 0)	N/A	N/A
100	69.3% (new 75)	74.8% (new 50)	95.4% (new 0)	N/A
125	67.3% (new 100)	74.8% (new 75)	82.3% (new 25)	N/A
150	66.1% (new 125)	75.2% (new 100)	80.4% (new 50)	N/A
180	64.4% (new 135)	72.2% (new 130)	77.3% (new 80)	93.95% (new 0)
200	66.0% (new 175)	73.6% (new 150)	77.8% (new 100)	89.2% (new 20)
210	65.9% (new 185)	73.1% (new 160)	77.6% (new 110)	87.2% (new 30)
220	64.8% (new 195)	73.4% (new 170)	73.8% (new 120)	81.8% (new 40)
230	68.8% (new 205)	74.6% (new 180)	75.8% (new 130)	84.3% (new 50)
250	65.1% (new 225)	72.2% (new 200)	74.3% (new 150)	84.2% (new 70)
268	66.5% (new 243)	74.6% (new 218)	76.6% (new 168)	84.5% (new 88)

Fig. 15. Chart showing the Siamese architecture performance on newly added cows

It was observed that the Siamese Network performed better compared to the Dot Product approach in terms of the stabilizing the model's performance while adding new cows. In comparison to the previous approach that achieved an accuracy of 84.1% when trained on 180 classes, the siamese network scores an accuracy of 93.95% on similar number of classes. However, the siamese network is slower compared to the dot product approach when it comes to prediction, as it takes an average of 14 seconds to predict the cow compared to the dot product with 1 second.

3) False Acceptance vs False Rejection Rate: Further experimentation were also done to evaluate the Siamese network False Acceptance Rate versus the False Rejection Rate over all 268 classes. The FAR, indicating instances where an incorrect class is accepted, increases significantly as the number of classes increases. Conversely, the FRR, reflecting instances where the correct class is rejected, remains relatively low and stable. As noted in the fig 16 below, it was observed that as the number of classes grows, the model becomes more prone to falsely accepting incorrect classes while maintaining a consistent rate of rejecting correct classes.

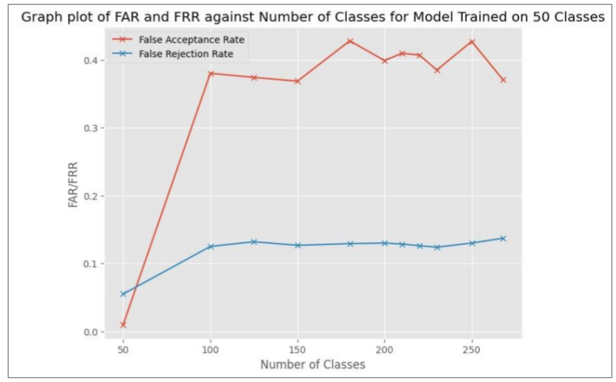

Fig. 16. Chart showing the FAR and FRR of the Siamese Network

V. Recommendation and Future Work

In this project, it was observed that the siamese network and the ResNet50 model with the dot product both achieved the primary objectives of identifying a cow given its muzzle print with a good accuracy while also accommodating new cows without the need to retrain on all images. It was also noted that the dot product accuracy depends on the quality of images used in feature representation, thus the reason behind a good accuracy with field data while not performing well on the original transformed data. Also due to their unique architectures, the siamese network works better on image verification while the dot product works better on image identification.

For future related work, we recommend a consolidation of a bigger and more representative dataset containing diverse cow muzzle print images to use in experimenting the two researched architectures. Additionally, there is a need to explore other similarity evaluation metrics to the dot product, such as Cosine Similarity, and Nearest Neighbors, among others.

References

[1] S. Kumar, S. K. Singh, R. Singh, and A. K. Singh, "Animal Biometrics: Concepts and Recent Application," en, in *Animal Biometrics: Techniques and Applications*, S. Kumar, S. K. Singh, R. Singh, and A. K. Singh, Eds., Singapore: Springer, 2017, pp. 1–20, ISBN: 978-981-10-7956-6. DOI: 10.1007/978-981-10-7956-6_1. [Online]. Available: https://doi.org/10.1007/978-981-10-7956-6_1 (visited on 12/15/2023).

[2] K. Shaheed, A. Mao, I. Qureshi, *et al.*, "A Systematic Review on Physiological-Based Biometric Recognition Systems: Current and Future Trends," en, *Archives of Computational Methods in Engineering*, vol. 28, no. 7, pp. 4917–4960, Dec. 2021, ISSN: 1886-1784. DOI: 10.1007/s11831-021-09560-3. [Online]. Available: https://doi.org/10.1007/s11831-021-09560-3 (visited on 09/29/2023).

[3] Y. Xiong, G. Li, and G. Erickson, *Beef Cattle Muzzle/Noseprint database for individual identification*, Mar. 2022. DOI: 10.5281/zenodo.6324361. [Online]. Available: https://zenodo.org/records/6324361 (visited on 12/15/2023).

[4] K. Shaheed, L. Yang, G. Yang, I. Qureshi, and Y. Yin, "Novel Image Quality Assessment and Enhancement Techniques for Finger Vein Recognition," in *2018 International Conference on Security, Pattern Analysis, and Cybernetics (SPAC)*, Dec. 2018, pp. 223–231. DOI: 10.1109/SPAC46244.2018.8965537. [Online]. Available: https://ieeexplore.ieee.org/document/8965537 (visited on 12/13/2023).

[5] J. A. Unar, W. C. Seng, and A. Abbasi, "A review of biometric technology along with trends and prospects," *Pattern Recognition*, vol. 47, no. 8, pp. 2673–2688, Aug. 2014, ISSN: 0031-3203. DOI: 10.1016/j.patcog.2014.01.016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S003132031400034X (visited on 12/13/2023).

[6] R. D. Chaudhari, A. A. Pawar, and R. S. Deore, "The Historical Development Of Biometric Authentication Techniques: A Recent Overview," en, *International Journal of Engineering Research*, vol. 2, no. 10, 2013.

[7] T. Burghardt, "Visual Animal Biometrics," 2008. DOI: https://doi.org/10.1016/j.future.2016.12.006.

[8] A. I. Khan and M. A. Wani, "Patch-based Segmentation of Latent Fingerprint Images Using Convolutional Neural Network," *Applied Artificial Intelligence*, vol. 33, no. 1, pp. 87–100, Jan. 2019, Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/08839514.2018.1526704, ISSN: 0883-9514. DOI: 10.1080/08839514.2018.1526704. [Online]. Available: https://doi.org/10.1080/08839514.2018.1526704 (visited on 12/15/2023).

[9] S. Minaee and A. Abdolrashidi, *DeepIris: Iris Recognition Using A Deep Learning Approach*, arXiv:1907.09380 [cs], Jul. 2019. DOI: 10.48550/arXiv.1907.09380. [Online]. Available: http://arxiv.org/abs/1907.09380 (visited on 12/15/2023).

[10] K. Nguyen, C. Fookes, R. Jillela, S. Sridharan, and A. Ross, "Long range iris recognition: A survey," *Pattern Recognition*, vol. 72, pp. 123–143, Dec. 2017, ISSN: 0031-3203. DOI: 10.1016/j.patcog.2017.05.021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0031320317302182 (visited on 12/15/2023).

[11] J. M. Song, W. Kim, and K. R. Park, "Finger-Vein Recognition Based on Deep DenseNet Using Composite Image," *IEEE Access*, vol. 7, pp. 66 845–66 863, 2019, Conference Name: IEEE Access, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2019.2918503. [Online]. Available: https://ieeexplore.ieee.org/document/8721065 (visited on 12/15/2023).

[12] I. Joshi, A. Anand, M. Vatsa, R. Singh, S. D. Roy, and P. Kalra, "Latent Fingerprint Enhancement Using Generative Adversarial Networks," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, ISSN: 1550-5790, Jan. 2019, pp. 895–903. DOI: 10.1109/WACV.2019.00100. [Online]. Available: https://ieeexplore.ieee.org/document/8658707 (visited on 12/15/2023).

[13] J.-J. Lv, X.-H. Shao, J.-S. Huang, X.-D. Zhou, and X. Zhou, "Data augmentation for face recognition," *Neurocomputing*, vol. 230, pp. 184–196, Mar. 2017, ISSN: 0925-2312. DOI: 10.1016/j.neucom.2016.12.025. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231216315016 (visited on 12/15/2023).

[14] K. Sundararajan and D. L. Woodard, "Deep Learning for Biometrics: A Survey," *ACM Computing Surveys*, vol. 51, no. 3, 65:1–65:34, May 2018, ISSN: 0360-0300. DOI: 10.1145/3190618. [Online]. Available: https://doi.org/10.1145/3190618 (visited on 12/15/2023).

[15] Z. Li, Y. Wang, Z. Yang, *et al.*, "A novel fingerprint recognition method based on a Siamese neural network," en, *Journal of Intelligent Systems*, vol. 31, no. 1, pp. 690–705, Jan. 2022, Publisher: De Gruyter, ISSN: 2191-026X. DOI: 10.1515/jisys-2022-0055. [Online]. Available: https://www.degruyter.com/document/doi/10.1515/jisys-2022-0055/html?lang=en (visited on 09/29/2023).

[16] H. S. Kühl and T. Burghardt, "Animal biometrics: Quantifying and detecting phenotypic appearance," *Trends in Ecology & Evolution*, vol. 28, no. 7, pp. 432–441, Jul. 2013, ISSN: 0169-5347. DOI: 10.1016/j.tree.2013.02.013. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0169534713000657 (visited on 12/14/2023).

[17] D. Petrovska-Delacrétaz, A. Edwards, J. Chiassoli, G. Chollet, and D. S. Pilliod, "A reference system for animal biometrics: Application to the northern leopard frog," in *2014 1st International Conference on Advanced Technologies for Signal and Image Processing (ATSIP)*, Mar. 2014, pp. 295–300. DOI: 10.1109/ATSIP.2014.6834623. [Online]. Available: https://ieeexplore.ieee.org/document/6834623 (visited on 12/14/2023).

[18] A. Taha, A. Darwish, and A. E. Hassanien, "Arabian Horse Identification System Based on Live Captured Muzzle Print Images," en, in *Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2017*, A. E. Hassanien, K. Shaalan, T. Gaber, and M. F. Tolba, Eds., ser. Advances in

Intelligent Systems and Computing, Cham: Springer International Publishing, 2018, pp. 778–787, ISBN: 978-3-319-64861-3. DOI: 10.1007/978-3-319-64861-3_73.

[19] Y.-Y. Nguwi, A. Z. Kouzani, J. J. Kumar, and D. Driscoll, "Automatic detection of lizards," in *2016 International Conference on Advanced Mechatronic Systems (ICAMechS)*, ISSN: 2325-0690, Nov. 2016, pp. 300–305. DOI: 10.1109/ICAMechS.2016.7813464. [Online]. Available: https://ieeexplore.ieee.org/document/7813464 (visited on 12/14/2023).

[20] S. Kumar and S. K. Singh, "Visual animal biometrics: Survey," en, *IET Biometrics*, vol. 6, no. 3, pp. 139–156, 2017, ISSN: 2047-4946. DOI: 10.1049/iet-bmt.2016.0017. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1049/iet-bmt.2016.0017 (visited on 12/15/2023).

[21] B. Cao, Y. Kang, S. Lin, *et al.*, "A novel 3D model retrieval system based on three-view sketches," *Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology*, vol. 31, no. 5, pp. 2675–2683, Jan. 2016, ISSN: 1064-1246. DOI: 10.3233/JIFS-169107. [Online]. Available: https://doi.org/10.3233/JIFS-169107 (visited on 12/15/2023).

[22] T. Ahonen, A. Hadid, and M. Pietikainen, "Face Description with Local Binary Patterns: Application to Face Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2037–2041, Dec. 2006, ISSN: 0162-8828. DOI: 10.1109/TPAMI.2006.244. [Online]. Available: https://doi.org/10.1109/TPAMI.2006.244 (visited on 12/15/2023).

[23] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, Jul. 2002, Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence, ISSN: 1939-3539. DOI: 10.1109/TPAMI.2002.1017623. [Online]. Available: https://ieeexplore.ieee.org/document/1017623 (visited on 12/15/2023).

[24] D. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 2, Sep. 1999, 1150–1157 vol.2. DOI: 10.1109/ICCV.1999.790410. [Online]. Available: https://ieeexplore.ieee.org/document/790410/authors#authors (visited on 12/15/2023).

[25] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-Up Robust Features (SURF)," *Computer Vision and Image Understanding*, Similarity Matching in Computer Vision and Multimedia, vol. 110, no. 3, pp. 346–359, Jun. 2008, ISSN: 1077-3142. DOI: 10.1016/j.cviu.2007.09.014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1077314207001555 (visited on 12/15/2023).

[26] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," en, in *Proceedings of the 25th international conference on Machine learning - ICML '08*, Helsinki, Finland: ACM Press, 2008, pp. 1096–1103, ISBN: 978-1-60558-205-4. DOI: 10.1145/1390156.1390294. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1390156.1390294 (visited on 12/15/2023).